

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS
INDUSTRIALES Y DE TELECOMUNICACIÓN

UNIVERSIDAD DE CANTABRIA



Trabajo Fin de Máster

Desarrollo de un software de medida avanzada de
factor de calidad de cavidades resonantes
para aceleradores de partículas

Development of an advanced resonant cavity quality
factor measurement software for particle accelerators

Para acceder al Título de

Máster Universitario en Ingeniería de Telecomunicación

Autor: Edurne Monteoliva de la Pedraja
Septiembre, 2021

Resumen

El trabajo fin de máster, realizado en colaboración con el Consorcio ESS Bilbao, ha sido un trabajo en equipo donde he participado, en coordinación con un compañero, en dos proyectos.

El proyecto sobre el que trata este trabajo fin de máster es el desarrollo de un software de medida avanzada de factor de calidad de cavidades resonantes para aceleradores de partículas. Se realiza utilizando el lenguaje de programación Python. Este proyecto se lleva a cabo ante la importancia de un alto factor de calidad en las cavidades resonantes que forman parte de un acelerador de partículas, ya que esto implica una baja tasa de pérdidas de energía en relación con la energía almacenada “útil” en el resonador. El programa realiza el cálculo del factor de calidad utilizando varios métodos matemáticos diferentes. El método de los tres puntos es el más sencillo de los tres y permite una aproximación rápida y bastante realista. Los otros dos métodos, propuestos por D. Kajfez y K. Leong, están basados en interpolaciones (ajuste de curvas), por tanto, son más robustos ante los inevitables errores aleatorios de medida y el resultado que se obtiene es mucho más cercano a su valor real. Además, el software cuenta con una interfaz gráfica de usuario a través de la cual se indica si el cálculo del factor de calidad se va a realizar bien utilizando un analizador vectorial de redes en línea o bien utilizando un fichero de resultados de parámetros S (S2P) fuera de línea. La interfaz de usuario también presenta el resultado de factor de calidad final obtenido, es decir, el más exacto para las condiciones de medida. Para comprobar que el programa se había realizado correctamente, se realizaron comparaciones con simulaciones de resonadores (cuyo valor del factor de calidad era conocido) y medidas en laboratorio del factor de calidad en prototipos de resonadores y en una de las cavidades resonantes de tipo “Buncher” disponibles en ESS Bilbao.

A modo de resumen, destacaría que se ha tratado de un trabajo realizado en equipo que ha incluido un análisis de la bibliografía técnica relevante, el desarrollo de software en Python (un lenguaje de programación ampliamente utilizado en ciencia) y el manejo tanto manual como controlado remotamente de instrumentos de medida de radiofrecuencia especializados, con el objetivo de realizar medidas avanzadas de caracterización de cavidades resonantes para aceleradores de partículas.

Abstract

The final master's thesis, carried out in collaboration with the ESS Bilbao Consortium, has been a team effort where I have participated, in coordination with a colleague, in two projects.

The project, development of a software for advanced measurement of the quality factor of resonant cavities for particle accelerators, is the one that is going to be dealt with in this master's thesis. It is done using the Python programming language. This project is carried out due to the importance of a high quality factor in the resonant cavities that are part of a particle accelerator, since this implies a low rate of energy losses in relation to the "useful" stored energy in the resonator. The program performs the calculation of the quality factor using several different mathematical methods. The three-point method is the simplest of the three and allows a quick and fairly accurate approximation. The other two methods, proposed by D. Kajfez and K. Leong, are based on interpolations (curve fitting), therefore they are more robust to the inevitable random measurement errors and the result thus obtained is much closer to its true value. In addition, the software has a graphical user interface through which it is indicated whether the calculation of the quality factor is going to be carried out either using an online vector network analyzer or using an S parameter results file (S2P) off-line. The user interface also presents the final quality factor result, that is, the most accurate for the measurement conditions. To verify that the program had been carried out correctly, comparisons were made with resonator simulations (whose value of the quality factor was known) and laboratory measurements of the quality factor in prototype resonators and in one of the resonant cavities of the "Buncher" type available at ESS Bilbao.

As a summary, I would highlight that it has been a teamwork that has included an analysis of the relevant technical bibliography, the development of software in Python (a programming language widely used in science) and both manual and remote control of specialized radiofrequency measurement instrumentation, with the aim of carrying out advanced characterization measurements of resonant cavities for particle accelerators.

Agradecimientos

Me gustaría comenzar agradeciendo en primer lugar a ESS Bilbao por brindarme esta oportunidad, en particular a Pedro por su paciencia, ayuda y guía durante estos meses de trabajo. También agradecerle a Nagore e Ibon por su ayuda y disposición las veces que se ha acudido a ESS Bilbao para realizar las diferentes pruebas.

En segundo lugar, agradecerle a Tomás la paciencia infinita que ha tenido durante el trabajo de fin de máster, la ayuda que siempre que ha podido nos ha brindado y por guiarnos y tutelarnos junto a Pedro en este proyecto.

Finalmente, me gustaría agradecer, a mis padres y mis hermanos, el apoyo incondicional y paciencia que han tenido conmigo durante estos dos últimos años. Por último, a los que ya no están pero que seguro que están orgullosos de que termine esta etapa en mi vida.

Índice

Resumen	2
Abstract	3
Agradecimientos	4
Índice	5
Índice de figuras	6
Capítulo 1: Introducción	8
1.1 Acelerador de partículas	8
1.1.1 Acelerador de partículas lineal	8
1.1.2 Acelerador de partículas circular	9
1.1.3 CERN (Organización europea para la investigación nuclear)	10
1.2. ESS Bilbao	13
1.3. Aplicación médica del acelerador de partículas	16
Capítulo 2	18
2.1. Introducción a las cavidades resonantes	18
2.1.1 Cavity Buncher	19
2.1.2 Cavity RFQ	20
2.2. Medidas de caracterización de cavidades	23
2.2.1. Modos resonantes	23
2.2.2. Factor de Calidad	27
2.2.3. Factor de Acoplo	29
2.2.4. R/Q	31
2.3. Medida del Factor de calidad	34
2.3.1. Métodos de los 3 puntos	34
2.3.2. Método de Kajfez	37
2.3.3. Método de K. Leong	44
Capítulo 3: Desarrollo	51
3.1. Medida automatizada del factor de calidad en cavidades resonantes	51
3.2. Fabricación de un prototipo de resonador	60
Capítulo 4: Resultados	68
Capítulo 5: Conclusión y líneas futuras	80
Bibliografía	82
ANEXO: Código fuente	84

Índice de figuras

FIGURA 1. ESQUEMA DEL LINAC SIMPLIFICADO	8
FIGURA 2. ACELERADOR DE PARTÍCULAS CIRCULAR [3]	10
FIGURA 3. RELACIÓN ENTRE LAS PARTÍCULAS Y EL BOSÓN DE HIGGS.	12
FIGURA 4. LOCALIZACIÓN DEL MEBT	13
FIGURA 5. ESQUEMA DEL FUNCIONAMIENTO DE UNA CAVIDAD RESONANTE [7]	19
FIGURA 6. CAVIDAD BUNCHER DE ESS-BILBAO.....	20
FIGURA 7. RFQ FABRICADO PARA ESS-BILBAO	21
FIGURA 8. MEBT QUE ESS-BILBAO ENTREGARÁ A ESS LUND.	22
FIGURA 9. CIRCUITO EQUIVALENTE PARA UNA LÍNEA DE TRANSMISIÓN [11].....	24
FIGURA 10. DISTRIBUCIÓN DE LOS CAMPOS PARA LOS MODOS DE LA CAVIDAD PILLBOX.	25
FIGURA 11. DISTRIBUCIÓN DE LOS CAMPOS PARA LOS MODOS MONOPOLO.	25
FIGURA 12. DISTRIBUCIÓN DE LOS CAMPOS PARA LOS MODOS DIPOLO.	26
FIGURA 13. COEFICIENTES DE REFLEXIÓN DEL RESONADOR	30
FIGURA 14. PERTURBACIÓN EN UN CAMPO UNIFORME UTILIZANDO UNA PERLA DIALÉCTRICA.....	31
FIGURA 15. CÁLCULO DE LA Q UTILIZANDO EL MÉTODO DE LOS TRES PUNTOS.	36
FIGURA 16. CIRCUITO RESONADOR ACOPLADO A UN CIRCUITO EXTERNO DE DOS PUERTOS SIN PÉRDIDAS [14].	38
FIGURA 17. COEFICIENTE DE REFLEXIÓN A LA ENTRADA EN FUNCIÓN DE LA FRECUENCIA, EN EL CIRCUITO SIN PÉRDIDAS ACOPLADO.	40
FIGURA 18. CIRCUITO EQUIVALENTE DONDE R_s VALE CERO	42
FIGURA 19. TÍPICO LAZO DE ACOPLO	43
FIGURA 20. MODELO DEL CIRCUITO EN TRANSMISIÓN DEL SISTEMA DE UN RESONADOR.	45
FIGURA 21. CIRCUITO EQUIVALENTE EN TRANSMISIÓN DEL RESONADOR CON PÉRDIDAS EN ACOPLO CON PÉRDIDAS [15].....	48
FIGURA 22. DIAGRAMA DE FLUJO DEL PROGRAMA DE MEDIDA DE FACTOR DE CALIDAD	51
FIGURA 23. INTERFAZ DE USUARIO	52
FIGURA 24. S_{11} A LA FRECUENCIA DE RESONANCIA Y S_{11} DESINTONIZADO.	54
FIGURA 25. INTERPOLACIÓN REALIZADA EN PYTHON	55
FIGURA 26. RESOLUCIÓN DEL SISTEMA EN "QU_REFL"	55
FIGURA 27. BÚSQUEDA DE LOS ÍNDICES	57
FIGURA 28. INTERPOLACIÓN.	57
FIGURA 29. CÁLCULO DE LAS BETAS EN FUNCIÓN DE SI ES SOBRE/INFRA ACOPLADO. ...	58
FIGURA 30. FUNCIÓN NP.WHERE	58
FIGURA 31. LLAMADA A "DEF QU_REFL" PARA CALCULAR LOS DIÁMETROS NECESARIOS.	59
FIGURA 32. RESULTADO DEL FACTOR DE CALIDAD MEDIDO PARA UN RESONADOR.....	60
FIGURA 33. CIRCUITO CON COMPONENTES IDEALES.	61
FIGURA 34. RESULTADO DE LA SIMULACIÓN DEL CIRCUITO IDEAL.....	61
FIGURA 35. CIRCUITO RESULTANTE DE UTILIZAR LA HERRAMIENTA "TUNE"	62
FIGURA 36. RESULTADO DE LA SIMULACIÓN DEL CIRCUITO TRAS UTILIZAR LA HERRAMIENTA "TUNE"	62
FIGURA 37. DEFINICIÓN DEL SUSTRATO.	63
FIGURA 38. ESQUEMÁTICO CORRESPONDIENTE AL RESONADOR REAL.	63
FIGURA 39. SIMULACIÓN COMPONENTES REALES.	64
FIGURA 40. SIMULACIÓN CON COMPONENTES REALES	64
FIGURA 41. SIMULACIÓN CON COMPONENTES REALES	65
FIGURA 42. LAYOUT FINAL	66
FIGURA 43. RESONADOR FABRICADO	66
FIGURA 44. CONDENSADOR EN PARALELO CON EL PRIMER GAP.	67
FIGURA 45. RESONADOR A 361 MHZ.	68
FIGURA 46. MEDIDA DEL FACTOR DE CALIDAD CON EL VNA.	69

FIGURA 47. REPRESENTACIÓN S_{11} Y S_{22} EN CARTA DE SMITH.	70
FIGURA 48. MEDIDA DEL FACTOR DE CALIDAD DEL RESONADOR A 361 MHZ.	70
FIGURA 49. CÁLCULOS DEL FACTOR DE CALIDAD DEL RESONADOR A 361 MHZ.	71
FIGURA 50. CONEXIÓN DE LOS PUERTOS AL ANALIZADOR.....	72
FIGURA 51. CAVIDAD BUNCHER CON FACTOR DE ACOPLO 1 Y 0.....	73
FIGURA 52. RESULTADO ÓPTIMO DEL FACTOR DE CALIDAD MOSTRADO EN LA INTERFAZ DE USUARIO.	73
FIGURA 53. RESULTADOS PARA LA PRIMERA MEDIDA DE LA CAVIDAD BUNCHER.	74
FIGURA 54. MEDIDA CAVIDAD BUNCHER β_1 MAYOR QUE 1 Y β_2 MAYOR QUE 0.	75
FIGURA 55. RESULTADO ÓPTIMO DE LA CAVIDAD BUNCHER SOBRE ACOPLADA.	75
FIGURA 56. TODOS LOS RESULTADOS OBTENIDOS POR EL PROGRAMA PARA UNA β_1 MUCHO MAYOR QUE 1.....	76
FIGURA 57. CAVIDAD BUNCHER CON UNA β_1 MENOR QUE 1.....	77
FIGURA 58. REPRESENTACIÓN RESULTADOS EN LA INTERFAZ DE USUARIO PARA UNA β_1 MENOR QUE 1.....	77
FIGURA 59. RESULTADOS DE LOS FACTORES DE CALIDAD PARA UNA β_1 MENOR QUE 1	78

Capítulo 1: Introducción

1.1 Acelerador de partículas

Un acelerador de partículas es un dispositivo que utiliza campos electromagnéticos para acelerar partículas cargadas (iones) a altas velocidades [1]. Muy a menudo, el haz de partículas utilizado está compuesto por protones o electrones, partículas subatómicas cargadas, incluso se utilizan átomos enteros de elementos como el oro o el uranio, que son mucho más pesados.

La aceleración de las partículas se puede usar para una gran variedad de propósitos [1]. En algunos casos, será con el objetivo de hacerlas colisionar con otras partículas, produciendo una gran cantidad de nuevas partículas cuyo estudio permite comprender mejor la estructura subatómica de la materia que nos rodea, el origen del universo, ... en definitiva, física fundamental. Asimismo, los aceleradores de partículas se emplean en medicina para investigación de nuevos fármacos y para diagnósticos y tratamientos, en ciencia de materiales, química, biología, energía y medio ambiente, o aplicaciones industriales, entre otras aplicaciones.

Por lo general, se distinguen dos tipos de básicos de aceleradores de partículas:

1.1.1 Acelerador de partículas lineal

El haz de electrones se crea típicamente [16] utilizando un acelerador de haz de electrones de alta energía, llamado acelerador lineal y a veces abreviado como **linac**. Un acelerador de partículas lineal permite aumentar la velocidad de iones o partículas subatómicas cargadas. Para ello se somete a las partículas cargadas a una serie de potenciales eléctricos oscilantes a lo largo del haz.

Los aceleradores de partículas lineales [16] se caracterizan por la función que presentan cada una de las secciones que forman parte de este. A continuación, se muestra un esquema de los componentes básicos que presenta un acelerador de partículas lineal, además de una explicación resumida de las secciones que le conforman:

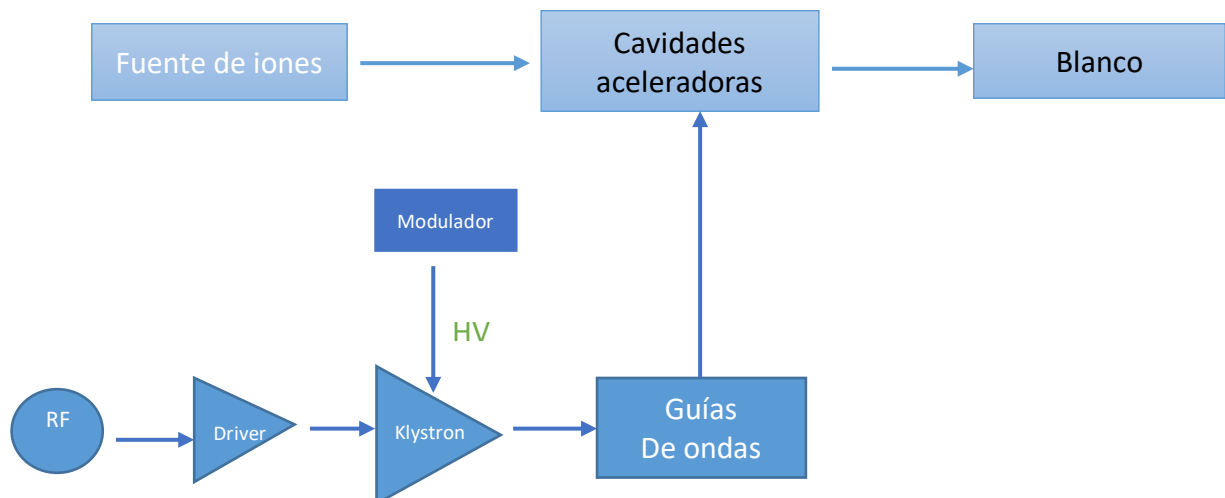


Figura 1. Esquema del LINAC simplificado

Como se ha comentado anteriormente y se puede observar en la figura anterior, existen varias secciones [16]. La primera es la sección se encarga de generar el haz (fuente de iones). La siguiente sección es la aceleradora y finalmente se encuentra la sección del blanco o target que es en la que se colisiona el haz de iones generado. Además, el acelerador de partículas lineal cuenta con un subsistema de radiofrecuencia, que está formado por generadores de RF, amplificadores de potencia, moduladores de alta tensión y guías de onda. Este sistema de RF es el encargado de alimentar a las cavidades resonantes aceleradoras del acelerador de partículas lineal.

- La fuente de iones: encargada de generar un haz de partículas. En muchos casos, en la fuente de iones se genera el plasma, en una cavidad que está sometida a alto vacío y donde se inyecta hidrógeno. En condiciones de resonancia ciclotrón del electrón (ECR, Electron Cyclotron Resonance), determinada por el campo magnético y la potencia de radiofrecuencia, se genera plasma de alta intensidad y después por diferencia de potencial electrostático se realizará la extracción de los iones (protones, por ejemplo).
- La cavidad aceleradora: es una cámara de metal. En su interior hay un alto vacío y presenta un campo electromagnético que oscila a una frecuencia de resonancia específica. Esta oscilación en el campo provoca la aceleración de las partículas.
- El generador de radiofrecuencia (RF): es quien genera una señal de un tono de alta frecuencia. Esta señal es alimentada a los varios componentes que forman parte del acelerador.
- El driver (preamplificador): amplificador de estado sólido. Se encarga de aumentar la amplitud de la señal de radiofrecuencia hasta alcanzar el valor de amplitud que requiere el Klystron a su entrada.
- Klystron (amplificador de potencia): es un tubo electrónico de vacío que proporciona altos niveles de potencia de RF. Su principio de funcionamiento es por modulación de la velocidad del haz de electrones en su interior.
- El modulador: es quien permite rectificar la corriente alterna de la red eléctrica de potencia, y generar pulsos de alta tensión que permiten alimentar al Klystron.
- Guías de onda: son las encargadas de transportar la potencia de radiofrecuencia ya amplificada e inyectarla en las cavidades aceleradoras por medio de un acoplador de potencia. Presentan pocas pérdidas de inserción en el conjunto.
- El blanco (target): elemento final del acelerador, hacia el que va dirigido el haz de iones.

1.1.2 Acelerador de partículas circular

En los aceleradores circulares [2], las partículas se mueven en una trayectoria casi circular, hasta que alcanzan la energía necesaria. Esta energía se consigue porque este tipo de aceleradores poseen una ventaja añadida y es que utilizan campos magnéticos además de campos eléctricos, lo que permite una aceleración mayor en espacios más

reducidos. Además, permiten que las partículas permanezcan confinadas en determinadas configuraciones teóricamente de manera indefinida.

Este tipo de aceleradores presenta un límite en la energía que pueden alcanzar. Esto ocurre debido a la radiación sincrotrón [2]. Esta radiación provoca una continua pérdida de energía, además de problemas que van asociados a este tipo de comportamiento.

Existen dos variantes de aceleradores circulares [3]: los ciclotrones (primer modelo de acelerador construido) y los sincrotrones, los cuales alcanzan energías del orden de TeV (billones de electrón-voltios)

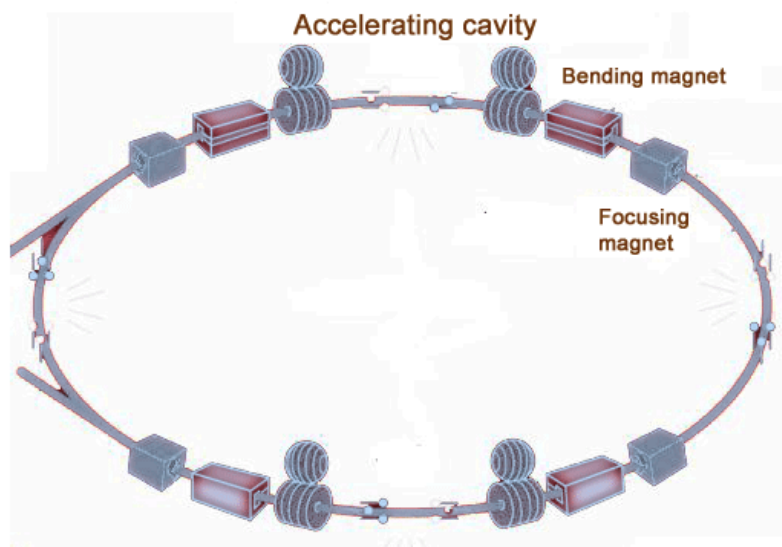


Figura 2. Acelerador de partículas circular [3]

La radiación que genera dicho acelerador es mayor cuanto más ligeras son las partículas. Por tanto, las partículas que se van a utilizar son habitualmente muy ligeras (principalmente electrones) cuando se pretende generar grandes cantidades de radiación. Aunque también se utilizan partículas pesadas, protones o núcleos ionizados más pesados, lo que permite que los aceleradores alcancen mayores energías. En el caso del CERN, se ha sustituido el colisionador de electrones y positrones (LEP) por el colisionador de hadrones (LHC)

1.1.3 CERN (Organización europea para la investigación nuclear)

A continuación, se va a hablar sobre el CERN y de su importancia en la investigación nuclear.

El CERN [4] está situado en la frontera franco-suiza, próximo a Ginebra. Es el mayor laboratorio de investigación en física de partículas elementales o física de altas energías a nivel mundial. Fue fundado en 1954 por doce países europeos, hoy en día es un modelo de colaboración científica internacional y uno de los centros de investigación más importantes del mundo.

En la actualidad el CERN [4] cuenta con 20 Estados miembros, los cuales comparten la financiación y la toma de decisión en la organización. El principal objetivo de la

investigación que lleva a cabo el CERN es el estudio de los constituyentes últimos de la materia. Como se ha comentado anteriormente, la tecnología utilizada para realizar este estudio ha sido los aceleradores de partículas.

Constituye un conjunto de aceleradores de partículas cuyo eslabón final es el **Gran Colisionador de Hadrones (LHC)**. El LHC está construido en un túnel de 27 Km de longitud, supone el mayor acelerador de partículas del mundo.

España ha pertenecido al CERN [4] entre 1962 y 1968, y luego desde 1983, cuando se incorporó de nuevo a la organización como miembro de plenos derechos. Cerca de 11.000 usuarios procedentes de 600 instituciones, departamentos universitarios y centros de investigación de todo el mundo participan en los programas de investigación del CERN. La presencia española representa entorno de un 5% en la categoría de usuarios reconocidos y registrados. A este número habría que añadir los casi 100 físicos teóricos que también colaboran con la organización.

Los distintos Estados contribuyen anualmente al CERN de manera proporcional al PIB de cada Estado. En el caso de España, hablando de contribución ocupa el quinto lugar por detrás de Alemania, Reino Unido, Francia e Italia.

Entre los hitos más importantes del CERN [7], ligados a descubrimientos científicos, podemos citar los siguientes:

- “La Partícula de Dios”: Se considera el descubrimiento más famoso de la organización, se ha podido realizar gracias al colisionador de Hadrones (LHC). En 2012, tras muchos años de teorías y experimentos, por fin, se pudo hallar el esquivo de bosón de Higgs, última pieza del modelo Estándar de la Física de partículas. Este descubrimiento ha supuesto un premio Nobel de Física en 2013 a Peter Higgs y François Englert, quienes predijeron la existencia del bosón. En la figura 1 se puede observar la relación entre el resto de las partículas y el bosón de Higgs.
- Las corrientes neutras débiles: Uno de los primeros grandes descubrimientos del CERN fue en 1973. Pudo llevarse a cabo gracias a una cámara de burbujas llamada Garmelle. Las corrientes neutras visibles son una manera en la que las partículas subatómicas interactúan entre sí y es una de las cuatro interacciones fundamentales de la física de las partículas. Sus precursores son Abdus Salam, Sheldon Glashow y Steven Weinberg que recibieron el premio Nobel en 1979.
- Los bosones W y Z: Una década más tarde del descubrimiento de las corrientes neutras débiles, en 1983, gracias al acelerador de partículas Super Proton Synchrotron (SPS, Súper Sincrotrón de Protones), los científicos descubrieron los bosones W y Z. Estas son las partículas que se medían en la fuerza débil. El bosón W tiene dos variantes con la misma masa, pero presenta cargas eléctricas opuestas (W^+ y W^-). Mientras que el Bosón Z no presenta carga. Se considera un descubrimiento esencial porque permitió poder avanzar en el Modelo Estándar y por ello sus descubridores (Carlo Rubbia y Simon Van der Meer) fueron premiados en 1984 con el premio Nobel.

- Los neutrinos ligeros: En 1989 se descubrieron los neutrinos ligeros. Son partículas elementales que apenas tienen carga o carecen de ella, además, raramente interactúan con otras partículas, lo que dificulta en gran medida su detención. Su descubrimiento se realizó utilizando el Large Electron-Positron Collider (LEP, Gran Colisionador de Electrones-Positrones), usando el detector ALEPH.
- La antimateria: La antimateria está compuesta por partículas que tienen la misma masa que una partícula de materia, pero presenta carga eléctrica opuesta (entre otras propiedades). Cuando la materia y la antimateria se combinan, se destruyen entre sí, lo que provoca que se eliminen grandes cantidades de energía. Es en 1995 cuando los científicos del CERN consiguen crear una forma de antimateria (anti-hidrógeno), ésta cuenta con una carga opuesta a la del hidrógeno. Sin embargo, la antimateria colisionó con la materia provocando su destrucción antes de que pudiese llevarse a cabo el estudio de esta. Fue en 2010 cuando se logró crear y aislar el anti-hidrógeno entorno a una sexta parte de un segundo. En 2011 fue cuando finalmente se consiguió aislar la antimateria en torno a 15 minutos.
- La violación de la simetría de paridad de cargas: Uno de los principales misterios de la cosmología es cómo puede ser que sabiendo que en el universo hay presencia de antimateria, exista la materia y que no se provoque la destrucción entre sí. Desde el punto de vista de las leyes físicas deben ser las mismas si una partícula se sustituye en una anti-partícula, esto recibe el nombre de simetría de paridad de carga. Pero los físicos nucleares **James Cronin** y **Val Fitch** del CERN demostraron que la paridad de carga es violada y por esa misma razón el universo existe. Este descubrimiento les otorgó el premio Nobel en 1980.

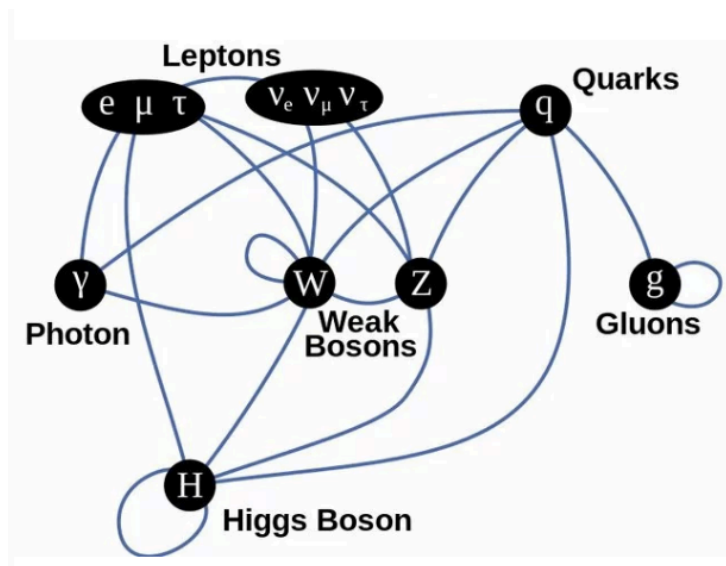


Figura 3. Relación entre las partículas y el Bosón de Higgs.

1.2. ESS Bilbao

El presente TFM se ha desarrollado dentro del marco de trabajo del Consorcio ESS Bilbao. A continuación, se va a presentar qué es el Consorcio ESS Bilbao y cuáles son sus objetivos y actividades.

El Consorcio ESS Bilbao [5] es un consorcio público de los gobiernos central y vasco. Su principal fin es aportar conocimiento y valor añadido en los campos de los aceleradores de partículas y las ciencias y tecnologías neutrónicas, a través de la contribución en especie a la Fuente Europea de Neutrones por Espalación, ESS (European Spallation Source).

La Fuente Europea de Neutrones por Espalación [16] es un consorcio europeo de infraestructura de investigación. El objetivo principal que presenta es diseñar, planificar, construir, poner en marcha, operar y dismantelar la fuente europea de espalación (ESS). Se podría resumir como un centro de investigación de materiales de gran escala. ESS va a construir y explotar la fuente de neutrones más potente del mundo y se pretende que sea uno de los centros líderes de investigación con neutrones en los campos químicos, la nanociencia, la energía, biociencia, la farmacéutica, la informática y la ciencia e ingeniería de materiales

La organización de ESS se lleva a cabo entre Suecia y Dinamarca [16]. En la actualidad, cuenta con la asociación de 15 países. ESS ERIC y los 15 países asociados son los encargados de realizar la construcción de ESS, se estima que esté finalizada en 2025. Los primeros neutrones que la instalación generará se espera que se produzcan en 2023 y que ya sea en 2025 cuando este en pleno funcionamiento con los 16 instrumentos que lo conforman.

Los proyectos en los que ESS Bilbao trabaja actualmente para su contribución a ESS Lund se muestran a continuación:

1. MEBT (Sección de Transporte de Haz de Media Energía)

ESS Bilbao [5] es responsable del diseño y fabricación del MEBT de ESS como contribución en especie. En la figura 4 se puede observar dónde se localiza el MEBT en el acelerador de partículas. En la sección de transporte de haz de energía media, las características del haz transversal y longitudinal se diagnostican y optimizan para generar una eficiente aceleración en el **Drift Tube Linac** (DTL).

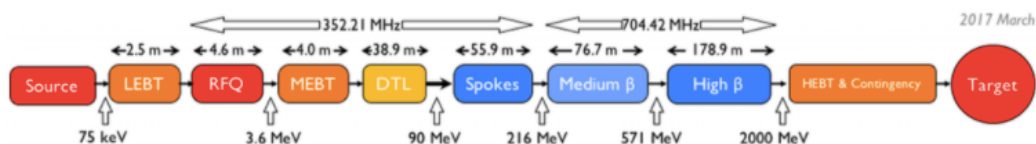


Figura 4. Localización del MEBT

Opera con un haz de protones pulsado de 3.6 MeV y corrientes de pico de hasta 62.5 mA, con pulsos nominales de 2.8 ms y una frecuencia de repetición de 14 Hz. El MEBT tiene una longitud de 3.81 m que permiten la integración de los diferentes componentes. Los componentes que forman parte del MEBT son

instalados en vasijas de vacío que se encuentran colocadas sobre un soporte muy complejo que permite la correcta alineación de todos los componentes.

Para adaptar las características del haz transversal se utilizan 11 cuádruplos, mientras que para el agrupamiento longitudinal se emplean tres cavidades “buncher”. Para formar el pulso de haz se utiliza un “chopper” rápido en tecnología “stripline” con tiempos de subida de menos de 10 ns. La caracterización del haz se realiza con un conjunto completo de diagnósticos: “wire scanner”, Unidad de Medida de Emitancia (EMU), Copa de Faraday, Monitores de Posición de Haz (“Beam Position Monitors” o BPMs), “Beam Current Monitors” (BCMs), ...

Con este proyecto se pretende alcanzar tres metas:

1. Hacer coincidir las características del haz de salida de RFQ con la entrada del DTL, tanto transversalmente usando los cuádruplos comentados, como longitudinalmente por medio de cavidades “buncher” de RF y modelado de pulsos con el Fast Chopper.
2. Caracterización del haz mediante un conjunto completo de diagnósticos.
3. Desarrollo del control e integración en el entorno EPICS (Experimental Physics and Industrial Control System)

2. Sistemas de RF

ESS [5] será la fuente de neutrones pulsados más potente e importante del mundo basada en un acelerador lineal (Linac) que acelera un haz pulsado de protones de 62.5 mA de corriente a un pico de energía de 2000 MeV.

El acelerador lineal de ESS incluye una sección de conducción normal o “caliente” compuesta por cavidades resonantes de RF basadas en cobre: un RFQ (Cuádrupolo de Radio Frecuencia), un MEBT (Transporte de haz de energía media) con tres cavidades tipo bunchers y 5 tanques DTL (Drift Tube Linac). Luego comienza la sección superconductora con las cavidades de tipo “Spoke”, para continuar con las cavidades elípticas de beta media y alta.

El linac conductor normal estará alimentado por nueve fuentes de energía de RF (una para la RFQ, tres para las cavidades “buncher” del MEBT y cinco para los tanques del DTL). Todas ellas operan a una frecuencia de 352.21 MHz y deberán proporcionar la energía de RF requerida para un haz de protones de 62.5 mA de corriente de pico, con una longitud de pulso de 2.86 ms y una tasa de repetición de pulsos de 14 Hz.

A partir de lo anteriormente visto, se puede definir el sistema de RF como todos los componentes y subsistemas necesarios para generar y entregar energía de RF a las distintas cavidades entre el “enchufe de la red eléctrica” y el acoplador de potencia de la cavidad. La cadena completa constará de moduladores de alta

tensión, amplificadores de RF, sistemas de distribución de RF, sistema de control de RF de bajo nivel (LLRF), sistemas de protección local y sistemas de control.

Con el sistema de RF se pretende alcanzar tres metas, las cuales se presentan a continuación:

1. Los sistemas de RF generan la potencia de RF necesaria para acelerar las partículas correctamente.
2. Los sistemas de RF tienen que mantener la amplitud y la fase del campo de aceleración dentro de una tolerancia dada para acelerar correctamente el haz de partículas cargadas sin degradar los parámetros de calidad de este.
3. Los sistemas de RF deben proteger a la cavidad y todos sus propios componentes de cualquier fallo.

3. Blanco (Target)

ESS Bilbao [5] está trabajando también en una contribución clave al **ESS Target Project** liderando varios de los componentes críticos. Es un proyecto que incluye todos los niveles de ingeniería nuclear, desde la física básica hasta la construcción y la instalación. El resultado final de este proyecto será una gran con más de 3000 toneladas de acero que podrá producir neutrones en el **Target**, reducir energía en el moderador y extraerla a los instrumentos de neutrones.

Con este proyecto se pretende alcanzar tres metas. Las cuales se presentan a continuación:

1. Suministro de componentes clave para la estación de blanco de ESS (incluyendo diseño conceptual, diseño de detalle, fabricación y ensamblaje).
2. Coordinar el proceso de diseño, incluyendo la inclusión de todas las interfaces relevantes con las partes interesadas.
3. Desarrollar conocimientos y capacidades técnicas en el diseño de componentes críticos en entornos de alta radiación para instalaciones científicas.
4. Producir haz de neutrones para realizar experimentos científicos en múltiples disciplinas.

4. Instrumento Miracles

Con este proyecto [5], se pretende que Miracles sea el espectrómetro de neutrones de retrodispersión de tiempo de vuelo de la fuente europea de espalación. ESS

Bilbao es el responsable principal de realizar el diseño, construcción, instalación y puesta en servicio del instrumento, junto a una colaboración con la Universidad de Copenhague y la Fuente Europea de Espalación.

Con este proyecto se pretende alcanzar tres metas. Las cuales se presentan a continuación:

1. Se pretende proporcionar un rango de transferencia de energía de 1 meV.
2. Rango dinámico de longitud de onda de aproximadamente 1,5 Angstrom (\AA) centrado en la línea elástica, con resolución de energía elástica variable.
3. La mejor resolución energética alcanzable, de $2 \mu\text{eV}$.

1.3. Aplicación médica del acelerador de partículas

En un principio, los aceleradores de partículas fueron diseñados y utilizados con un fin de investigación en el campo de la física nuclear y la física de altas energías. Actualmente, sus usos están siendo más diversos, como puede ser el campo de la ciencia de materiales, la biología, la medicina o incluso la arqueología.

En 1993 [6] se comenzaron con los tratamientos médicos con protones. En la actualidad se han tratado más de 50.000 pacientes irradiados en laboratorios de física subatómica y en más de diez centros de protonterapia situados en hospitales.

En estos centros donde se están tratando a los pacientes se utiliza bien un ciclotrón o un sincrotrón, ya que ambos tipos de aceleradores tienen la fiabilidad necesaria para operar en un entorno hospitalario.

Estos aceleradores tienen dos componentes principales: el sistema que se encarga de generar los iones, el sistema que distribuye el haz y el sistema de aplicación.

- Sistema encargado de producir iones: Se encarga de producir el haz protones. El ciclotrón de protones tiene una energía de 230 MeV y mediante un reductor de energía, necesario para realizar el tratamiento, se puede reducir hasta 70 MeV. Esta operación en un sincrotrón no es necesaria ya que la energía del haz de protones es variable.
- El sistema de distribución del haz quien se encarga de dirigir el haz de protones a la sala de tratamiento o de experimentación. Este sistema permite proporcionar un haz centrado, con las propiedades adecuadas para el experimento o el tratamiento pertinente.
- Sistema de aplicación: Contiene un **gantry**, que es una estructura de unos 10 m de altura que se puede girar $\pm 180^\circ$ alrededor del paciente y con el tamaño necesario que permite radiar al paciente desde cualquier ángulo. En la parte final del sistema, se puede encontrar un dispositivo que prepara y mide las

propiedades del haz y lo suministra de diferentes modos: dispersión simple o doble, barrido uniforme o en pincel.

El objetivo principal de cualquier sistema es producir y aplicar de manera correcta, en función del tratamiento, del haz tanto transversal como longitudinalmente.

En el tratamiento contra el cáncer se utiliza tanto protones como iones pesados (átomos completamente ionizados). El interés que presenta estos iones es que al tener mayor masa y carga que el protón, depositan una mayor energía por unidad de longitud de camino recorrido, lo que se traduce en una eficiencia mucho mayor en la curva de Bragg. En resumen, una mayor ionización produce una mayor efectividad. El objetivo principal en el tratamiento contra el cáncer, usando aceleradores de partículas, es atacar al ADN dentro de las células cancerígenas.

Capítulo 2

2.1. Introducción a las cavidades resonantes

En este punto se va a explicar la importancia de las cavidades resonantes en la aceleración del haz de partículas. En concreto se va a hablar con mayor profundidad de las cavidades Buncher y RFQ, que son las presentes en ESS-Bilbao y con las que se ha trabajado durante el TFM.

Las partículas cargadas se aceleran mediante campos eléctricos. Para ello existen dos formas: por corriente continua (DC) o por corriente alterna/radiofrecuencia (RF). Es en el caso de la corriente alterna donde se utilizan las cavidades resonantes.

La corriente alterna [7] acelera el haz invirtiendo la polaridad del campo eléctrico en cada semiperiodo de la sinusoide. Para conseguir esta aceleración, la partícula debe llegar en el momento preciso en que el campo presenta el sentido correcto para generar dicha aceleración. Esto implica que se debe cumplir la condición de sincronismo que se muestra en la fórmula a continuación:

$$T_{particula} = \frac{1}{2} T_{RF} \quad L = \frac{1}{2} vT \quad (1)$$

Donde:

- $T_{particula}$: periodo de la partícula
- T_{RF} : periodo de RF
- L : Separación entre elementos del acelerador
- V : velocidad de la partícula

Como se ha comentado anteriormente, las cavidades resonantes se utilizan a la hora de acelerar el haz de partículas utilizando radiofrecuencia. La energía de RF queda almacenada resonando en la cavidad. Dicha energía, producida por medio de generadores de RF de alta potencia (por ejemplo, tubos electrónicos de vacío como Klystrons o amplificadores de estado sólido, SSPA), es introducida en la cavidad a través de un acoplador. La frecuencia de resonancia de las cavidades puede variar con el tiempo, en función de la potencia disipada, la temperatura ambiente, etc. Las partículas están agrupadas en paquetes. En la figura 5 se muestra el esquema de funcionamiento de una cavidad resonante:

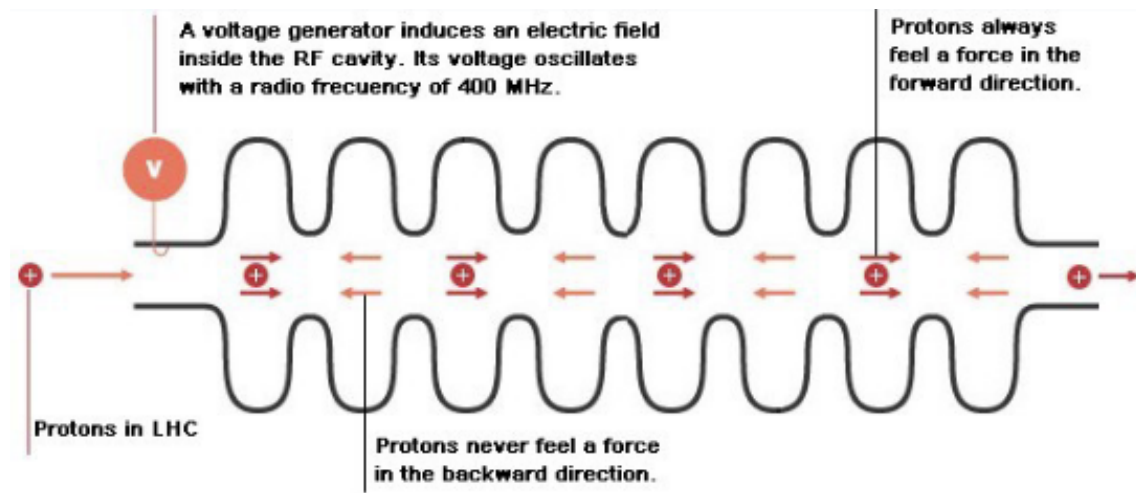


Figura 5. Esquema del funcionamiento de una cavidad resonante [7]

A continuación, se va a explicar las funcionalidades que presenta tanto la cavidad Buncher como el RFQ:

2.1.1 Cavidad Buncher

El objetivo principal de las cavidades de tipo Buncher es modificar el tamaño temporal de los paquetes de partículas aceleradas (Bunches). Para conseguir esta modificación se utiliza un campo de radiofrecuencia que permite agrupar las partículas en paquetes o “bunches” que se repiten con el período de la señal de radiofrecuencia. De esta forma, las cavidades de aceleración superconductoras, localizadas a continuación de las cavidades Buncher y DTL acelerarán los paquetes de partículas con mayor eficiencia.

La cavidad consta de una envolvente metálica [8] que hace las veces de cámara de vacío. Se trata habitualmente de piezas de cobre macizo que configuran la geometría interior de la cavidad y finalmente los campos eléctricos deseados en los gaps. Además, consta de una serie de elementos auxiliares. Los más importantes son los sintonizadores de frecuencia, el acoplador para la inyección de potencia de radiofrecuencia a la cavidad, la sonda de vacío, la refrigeradora líquida, las bombas de vacío y la sonda o pickup de medida del campo eléctrico en el interior de la cavidad.

En la figura 6 que se presenta a continuación se puede observar un prototipo de la cavidad Buncher con la que cuenta ESS-Bilbao:

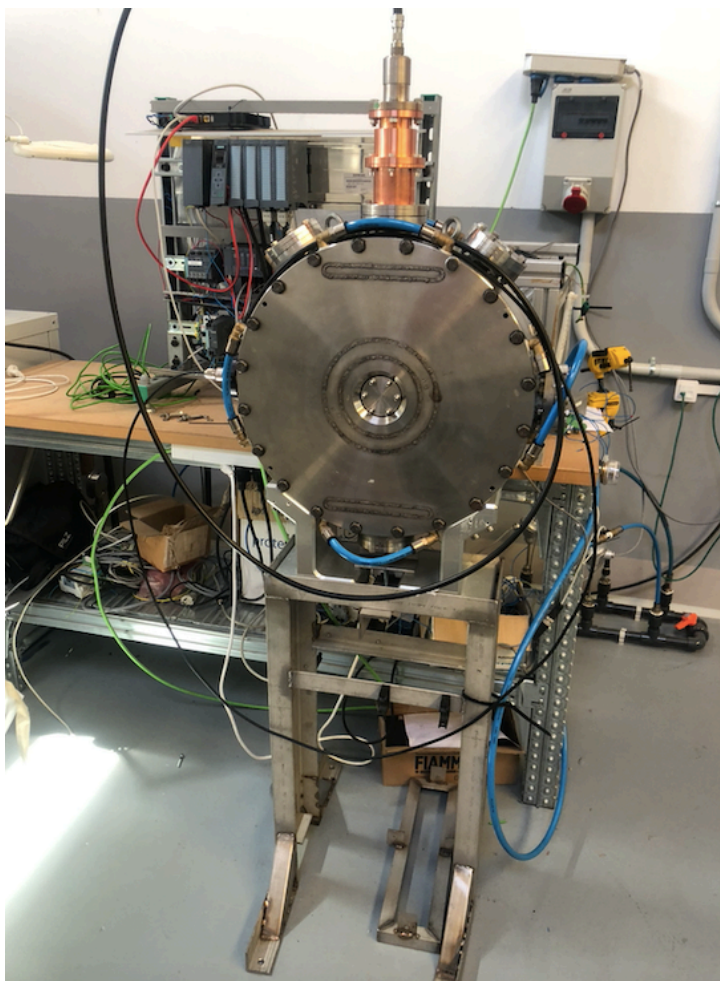


Figura 6. Cavity Buncher de ESS-Bilbao

2.1.2 Cavity RFQ

Un RFQ [9] (Radio Frequency Quadrupole) es una cavidad con una triple función, ya que es capaz de acelerar, enfocar y agrupar en paquetes un haz continuo de partículas cargadas con alta eficiencia. Todas estas características se obtienen de la distribución particular de campos eléctricos y magnéticos cuando la cavidad se excita con la frecuencia correspondiente al modo resonante adecuado (modo cuadrupolar). Todo ello se lleva a cabo sin necesidad de imanes para generar el campo magnético de enfoque. La geometría de la cavidad debe ser diseñada para hacerla resonar a la frecuencia de operación del acelerador. Las partículas son normalmente aceleradas desde decenas de keV hasta unos pocos MeV (por ejemplo, entre 3 y 5 MeV).

La invención del RFQ [9] ha permitido que se pueda preparar de manera eficiente un haz de hadrones de alta intensidad y baja energía extraído de una fuente de iones para la posterior aceleración en un Acelerador de Tubos de Deriva (Drift Tube Linac o DTL) convencional, aumentando la eficiencia de la transferencia entre la fuente y el DTL del 50% típico tradicionalmente a más del 90% en la actualidad.

El RFQ que se encuentra en ESS-Bilbao es una cavidad de unos 3.2 metros de largo, construida uniendo 4 segmentos de unos 800 mm. Cada uno de los segmentos está

constituido por 4 vanos, que forman los lóbulos y los electrodos cuyas características electromagnéticas permitirán alcanzar los altos requisitos necesarios.

En la figura que se presenta a continuación, figura 7, se puede observar el primer segmento de 800 mm del RFQ con el que cuenta ESS-Bilbao:

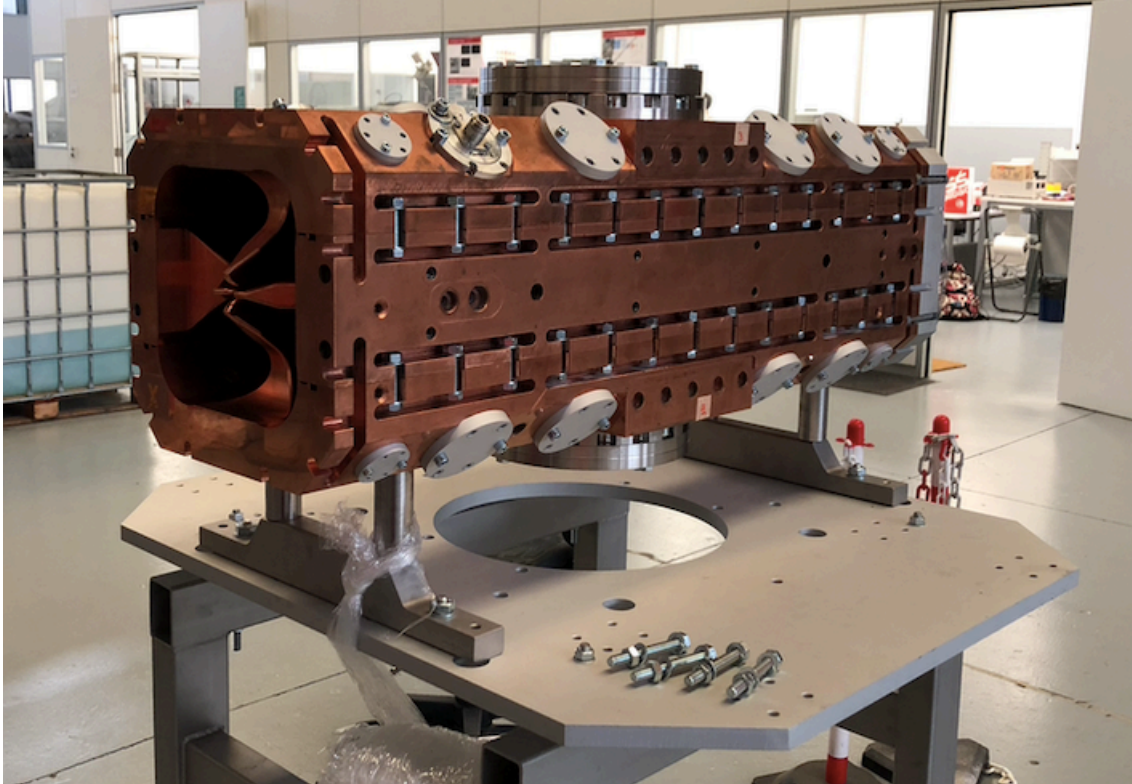


Figura 7. RFQ fabricado para ESS-Bilbao

En ESS-Bilbao tanto la cavidad Buncher como el RFQ son cavidades resonantes que se han desarrollado y fabricado y se están utilizando. A continuación, se muestra el trabajo de ESS con dichas cavidades:

- Como se ha comentado en puntos anteriores, ESS-Bilbao es el encargado de realizar el diseño y fabricación del MEBT (Sección de transporte de Haz a Media Energía) del acelerador de partículas de ESS. El MEBT además de contar con 11 cuadrupolos que permiten que las características transversales de haz coincidan cuenta también tres cavidades de RF Buncher que permiten cumplir con los parámetros longitudinales. En la siguiente figura (figura 8) se muestra el MEBT que ESS-Bilbao entregara a ESS en Lund.

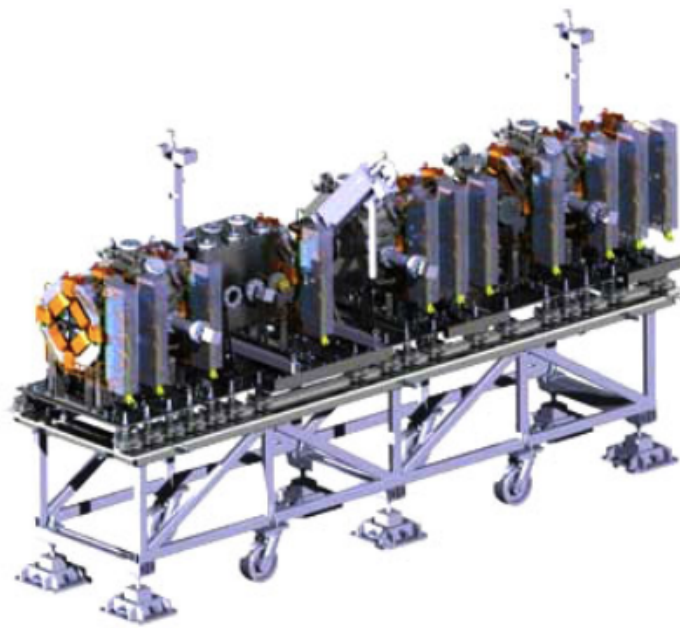


Figura 8. MEBT que ESS-Bilbao entregará a ESS Lund.

El RFQ se encuentra localizado justo antes del MEBT como se puede observar en la figura 4. El primer diseño que se hizo en ESS del RFQ fue en 2013. En 2015 tras una revisión y realizar diferentes cambios importantes se aprobó un nuevo diseño. La fabricación del primer segmento comenzó en 2016. En la figura 7, se puede observar el resultado final del primer segmento del RFQ en 2021, una vez que se han fabricado los 4 vanos que forman parte del mismo.

2.2. Medidas de caracterización de cavidades

Como se ha expuesto en el punto anterior, las cavidades de radiofrecuencia se utilizan para inyectar energía que permite acelerar el haz de partículas cargadas. En este punto se va a hacer una introducción a algunos de los principales parámetros característicos de las cavidades, como son los modos resonantes, factor de calidad, el acoplo, mapas de campos y R/Q.

2.2.1. Modos resonantes

Los campos electromagnéticos de las cavidades resonantes satisfacen las ecuaciones de Maxwell.

$$\left(\nabla^2 - \frac{1}{c^2} \frac{\partial^2}{\partial t^2} \right) \begin{Bmatrix} \vec{E} \\ \vec{H} \end{Bmatrix} = 0$$

Teniendo en cuenta las condiciones de contorno de las cavidades, es decir, que las paredes están hechas de materiales conductores de baja resistencia superficial, se cumple que:

- El campo eléctrico no presenta componente tangencial

$$\vec{n} \times \vec{E} = 0$$

El campo magnético no presenta componente

$$\vec{n} \cdot \vec{H} = 0$$

Suponiendo todo lo anterior, para una geometría de cavidad dada, las ecuaciones de Maxwell tienen un número infinito de soluciones con una dependencia del tiempo sinusoidal.

Para conseguir una aceleración eficiente, hay que escoger una geometría de cavidad y un modo que cumpla con los siguientes puntos:

- El campo eléctrico coincida con la trayectoria de la partícula.
- El campo magnético sea nulo a lo largo de la trayectoria de la partícula.
- La velocidad del campo electromagnético se corresponda con la velocidad que presenten las partículas.

Las cavidades presentan generalmente una gran cantidad de modos resonantes [11]. Las frecuencias se extienden hasta la frecuencia de corte (TE o TM). La resonancia suele estar dominada por el modo fundamental de la cavidad. A continuación, se presenta las diferentes formas de obtener una cavidad con modos:

- Línea de transmisión [11]: En las líneas de transmisión, se tiene en cuenta el circuito equivalente que aparece en la figura 9. Este circuito equivalente genera un patrón de campo. Si se cortase un extremo de la línea, tanto las ondas directas como las reflejadas generarían una onda estacionaria. Si se cortase el otro extremo de la línea entonces tendríamos una cavidad resonante.

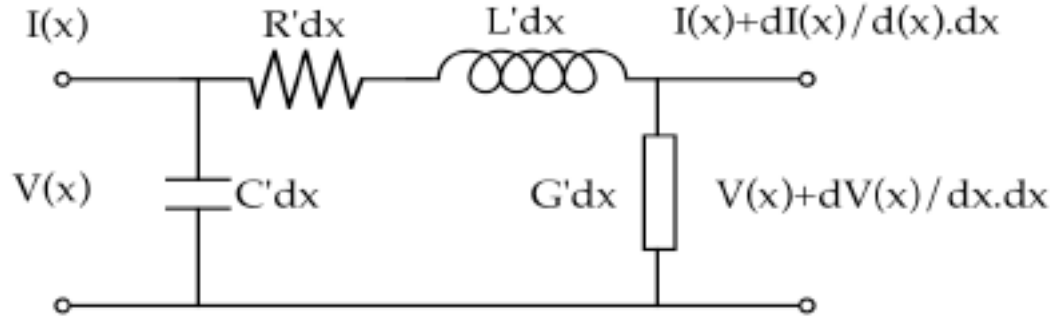


Figura 9. Circuito equivalente para una línea de transmisión [11].

- Cavidades cilíndricas (“Pillbox”) [11]: En este tipo de cavidades se consideran los dos primeros modos que se propagan en una guía de onda cilíndrica (TE = campo eléctrico transversal, TM = campo magnético transversal). En el caso de las guías de onda circulares, la frecuencia de corte está normalizada para para el modo más bajo (TE₁₁). Los únicos modos que presentan campo a lo largo de la dirección del haz y que puede interactuar con este, es el modo TM.

Las variaciones transversales del campo longitudinal son soluciones de las ecuaciones de Maxwell dentro de una condición circular. Además, las soluciones son funciones de Bessel del primer tipo. A continuación, se muestra los campos para los modos TM:

$$E_z(r, z, t, \phi) = E_0 J_m \left(\frac{x_{mn}}{a} r \right) e^{j\omega t} \cos(m\phi) \cos(K_z z) \quad (3)$$

$$H_\phi(r, z, t, \phi) = H_0 J_m \left(\frac{x_{mn}}{a} r \right) e^{j\omega t} \cos(m\phi) \cos(K_z z) \quad (4)$$

En la siguiente figura (figura 10) se puede observar las distribuciones para los modos en las cavidades Pillbox:

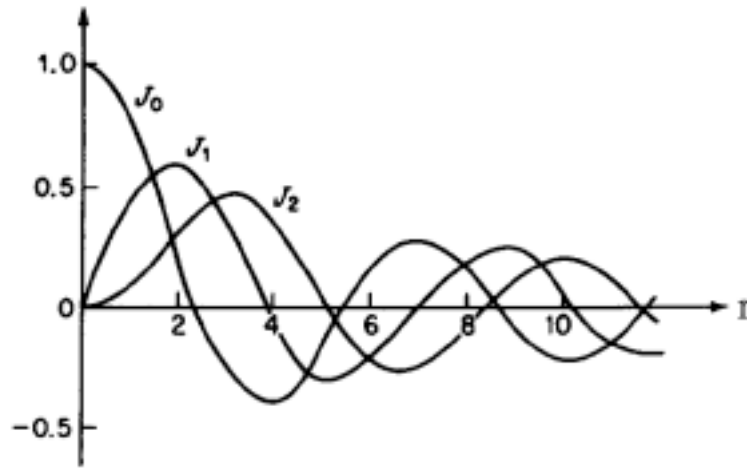


Figura 10. Distribución de los campos para los modos de la cavidad PillBox.

- Modos monopoles [11]: Los modos monopoles o modos TM son los modos que no presentan variación azimutal. Presentan un campo eléctrico longitudinal en el eje, por tanto, pueden interactuar con el haz de partículas.

La distribución radial que presenta E_z sigue a J_0 hasta que se hacen cero ($E_z = 0$) en la pared conductora con radio “a”. Algo similar ocurre con H_ϕ y E_r (en el caso de se presenten) que siguen a J'_0 y son cero en el centro y presentan un valor infinito en la pared.

En la siguiente figura (figura 11) se representa las distribuciones de los campos para los modos monopolo:

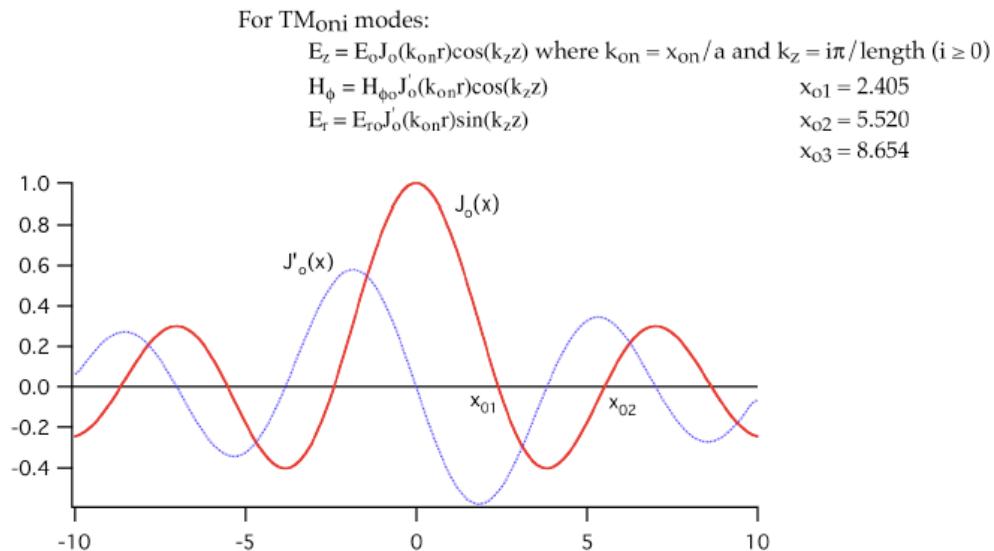
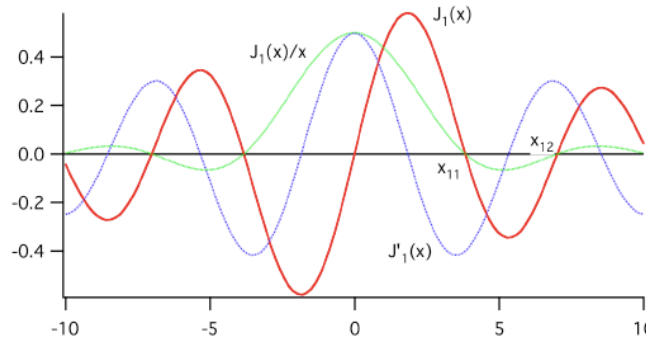


Figura 11. Distribución de los campos para los modos monopolo.

- Modos dipolares [11]: Estos modos se caracterizan por un periodo completo de variación alrededor del azimut. Esto en los modos TM significa que no hay campo longitudinal en el eje y que la intensidad del campo crece linealmente con un radio cercano al centro, con signo opuesto a cada lado del eje. Este gradiente transversal

al campo longitudinal da lugar a una subida de voltaje transversal que es proporcional a la corriente del haz y al desplazamiento del haz de partículas. Por tanto, se puede decir que no habrá una subida de voltaje si no hay gradiente de campo.

En la siguiente figura (figura 12) se representa las distribuciones de los campos para los modos dipolo:



For TM_{1ni} modes:

$$E_z = E_0 J_1(k_{1n}r) \cos(\phi) \cos(k_z z) \text{ where } k_{1n} = x_{1n}/a \text{ and } k_z = i\pi/\text{length} (i \geq 0)$$

$$H_\phi = H_{\phi 0} J_1'(k_{1n}r) \cos(\phi) \cos(k_z z) \quad x_{11} = 3.383171$$

$$|H_r| = H_{r0} \frac{J_1'}{r}(k_{1n}r) \sin(\phi) \cos(k_z z) \quad x_{12} = 7.01559$$

$$x_{13} = 10.17347$$

Figura 12. Distribución de los campos para los modos dipolo.

- Modos de orden superior ($m > 1$) [11]: Para modos que presentan un orden azimutal más alto ($m = 2$: cuadrupolo, $m = 3$: sextupolo, etc.) los campos cerca del eje del haz de partículas se vuelven progresivamente más débiles a medida que se concentra la energía almacenada hacia el borde exterior de la cavidad.

En el caso de los modos con “m” par, no hay inversión de signo a lo largo del eje y tienen un pequeño factor de tiempo de tránsito. Los modos con “m” impar pueden acoplarse débilmente al movimiento transversal del haz, generalmente cuando esto ocurre no suele generar ningún problema.

- Cavidades conductoras normales [11]: En este tipo de cavidades se parte de que $Q = \sqrt{\frac{L}{C}}$, donde L se maximiza y la C hay que minimizarla mientras se mantiene de manera optima y máxima la duración de la interacción (con un tiempo máximo de Tmax).

El “nose-cone” o cavidad reentrante permite: aumentar el volumen ocupado por el campo magnético, aumenta la superficie portadora de la corriente y desmulla la superficie con la región capacitiva.

Los factores que limitan el gradiente alcanzable son la disipación de energía de la pared y la fuerza del campo E en el “nose tips”.

- Cavidades súper conductoras [11]: La forma lisa que presenta es determinada por la necesidad de evitar la emisión de campo en la superficie. Presenta un factor de calidad muy alto. Los factores que limitan el gradiente máximo que se puede alcanzar son:
 - Emisión de impurezas desde la superficie del campo.
 - En última instancia, el gradiente está limitado por el campo magnético de la superficie.
- Sintonización de cavidades [11]: La frecuencia de resonancia del modo se puede ajustar introduciendo perturbaciones. El efecto dependerá en función de si la perturbación afecta a los campos eléctricos o magnéticos (cambio de capacitancia o inductancia).
- Cavidades BPMs [11]: Estas cavidades utilizan la dependencia lineal del campo excitado en el modo dipolo, como medida de la posición del haz. La medición completa (x,y) requiere polarizar cada modo, generalmente se consigue utilizando dos cavidades.

2.2.2. Factor de Calidad

El factor de calidad es uno de los parámetros más importantes a caracterizar en las cavidades aceleradoras, ya que habitualmente se requieren valores altísimos para reducir las pérdidas en las paredes de las mismas, incluso órdenes de magnitud mayores que los de los resonadores empleados en filtros y en osciladores (coaxiales, en guía de onda, dieléctricos...). No sólo se necesita maximizar el factor de calidad, sino que se debe conocer con la mayor exactitud posible su valor en términos absolutos, a fin de estimar la amplitud del campo acelerante a partir de la potencia de RF inyectada en la cavidad.

El factor de calidad es una relación entre la baja tasa de pérdidas de energía con la energía almacenada “útil” que presenta el resonador. En la teoría de circuitos elemental, el factor de calidad (factor Q) [12] del circuito resonante paralelo se define como:

$$Q_0 = \frac{\omega_0 C}{G_0} \quad (5)$$

Donde:

- Q_0 : es el factor de calidad descargado
- ω_0 : pulsación (rad/seg)
- G_0 : conductancia (S)
- C: capacidad (F)

Es una forma de relacionar la susceptancia (a la frecuencia de resonancia) y la conductancia [12] $G_0 = \frac{1}{R_0}$ del circuito. Cuanto mayor sea el factor Q, más estrecha será la curva en forma de campana, es decir, a mayor factor de calidad mayor es la selectividad frecuencial del circuito, por lo que el resonador discriminará mejor entre señales de diferentes frecuencias. De forma habitual, la letra Q es utilizada para designar la “calidad” del circuito.

A frecuencias de RF y de microondas, los resonadores rara vez se construyen en forma de inductancias y capacidades agrupadas. En cambio, consisten en circuitos distribuidos, como una línea de transmisión microstrip de media longitud de onda. En el caso de necesitar factores de calidad mucho más altos, se podrían alcanzar utilizando resonadores dieléctricos, que están integrados con circuitos microstrip.

En el caso de tener que manejar altas potencias, es necesario emplear cavidades cilíndricas o rectangulares huecas, como ocurre en el caso de los aceleradores de partículas, en el que la entrada está conectada a través de una línea de transmisión coaxial.

Cerca de la frecuencia de resonancia de interés [12], cada uno de los resonadores comentados tienen propiedades similares a las de un circuito resonador paralelo básico, es decir, la parte real de su impedancia de entrada exhibe una curva resonante en forma de campana, y el ancho de la curva (o ancho de banda) es inversamente proporcional al factor Q. Sin embargo, la definición de la teoría de circuitos del factor Q dada por la fórmula (5) no es aplicable directamente a circuitos resonadores de microondas. Por tanto, se podría decir que una definición más general del factor Q, válida tanto para resonadores de elementos concentrados como para resonadores de RF y microondas podría ser la siguiente:

$$Q = \left[\frac{\omega W_{max}}{P_d} \right]_{\omega = \omega_o} \quad (6)$$

Donde W_{max} es el pico máximo de energía almacenada en el resonador, P_d es la potencia media disipada en el resonador.

Se puede hablar de dos tipos de energía almacenada en cualquier resonador [12]: energía eléctrica (W_e) y energía magnética (W_m). Para el comportamiento armónico en el tiempo (seno y coseno) de los campos electromagnéticos, la energía en el resonador fluctúa entre los tipo eléctrico y magnético de tal manera que cuando W_e es máximo, W_m es cero y un cuarto de periodo después W_e es cero y W_m es máximo y así sucesivamente. Cuando el resonador está en funcionamiento a su frecuencia de resonancia, el valor de pico de W_e es el mismo que el valor de pico de W_m .

La distribución tridimensional de los campos eléctricos y magnéticos en el resonador debe determinarse mediante un procedimiento analítico o numérico. De esta manera es posible integrar la energía almacenada sobre el volumen del resonador y determinar la disipación de potencia en el resonador debido a pérdidas de conductor y/o pérdidas dieléctricas. Si se sustituyen estos valores en la ecuación (6) proporcionan el valor de factor Q descargado. El factor de calidad descargado es el factor de calidad en el que solo se tienen en cuenta las pérdidas exclusivas del circuito resonador, es decir, es el factor de calidad intrínseco de la cavidad.

Cuando el resonador se encuentra cargado presenta una frecuencia de resonancia ligeramente diferente, llamada frecuencia de resonancia cargada. Además, se reduce el valor de Q del conjunto a un nuevo valor llamado Q cargado, denotado como Q_L . El factor de calidad cargado es el factor de calidad que incluye todos los efectos de pérdidas, internas y externas, es decir, quien tiene en cuenta los factores de acoplo. Para poder calcular el valor de esta nueva Q hay que realizar el inverso a la fórmula (6) [12]:

$$\left[\frac{P_d}{\omega W_{max}} \right]_{\omega = \omega_L} = \frac{P_0}{\omega_L W_{max}} + \frac{P_{ex}}{\omega_L W_{max}} \quad (7)$$

En la fórmula anterior, la potencia disipada consta de dos partes: La potencia disipada en el resonador descargado (P_0) y la potencia disipada en el circuito externo (P_{ex}). Los tres términos de la fórmula (7) se pueden identificar cada uno como factores Q particulares [12]:

$$\frac{1}{Q_L} = \frac{1}{Q_0} + \frac{1}{Q_{ex}} \quad (8)$$

Donde Q_L es el factor general Q del resonador y su circuito externo. Q_0 es el factor Q descargado del resonador no perturbado y Q_{ex} es el factor Q externo. La relación entre la potencia disipada en el circuito externo y la potencia disipada en el resonador se denomina coeficiente de acoplo K. El acoplo k se puede definir por la siguiente fórmula:

$$K = \frac{V^2 G_{ex}}{V^2 G_0} = \frac{G_{ex}}{G_0} = \frac{Q_0}{Q_{ex}} \quad (9)$$

Si en la fórmula (8) se elimina la Q y se usa la fórmula (9) se obtiene la relación entre la Q cargada y la descargada de la siguiente manera:

$$Q_0 = Q_L(1 + k) \quad (10)$$

En el momento que se inicia una medida, el circuito externo (donde se localiza el analizador de redes) carga el resonador y la medida proporcionara la Q cargada (Q_L). Cuanto más fuerte sea el acoplo que se crea entre el analizador de redes y el resonador, menor será el valor de Q descargado. La medida debe realizarse de tal manera que además de proporcionar la Q cargada también debe proporcionar el acoplo para poder calcular el valor de Q descargado utilizando la fórmula (10).

Existen diferentes formas de medir el factor de calidad de un resonador. En este TFM se va a realizar un estudio del factor de calidad de tres maneras diferentes, las cuales se describirán detalladamente más adelante. A continuación, se muestra los tres métodos que se van a utilizar: El método de los 3 puntos (3dB) y los métodos propuestos por D. Kajfez y por K. Leong, los cuales presentan cierto parecido ya que están basados en ajuste de curvas.

2.2.3. Factor de Acoplo

El factor de acoplo es uno de los parámetros que se debe tener en cuenta en las cavidades resonantes. Su importancia deriva de que el factor de acoplo es la relación que existe entre la potencia que el circuito disipa al exterior con la potencia que el propio resonador disipa. En la figura que se muestra a continuación, se puede observar que para medir este parámetro hay que caracterizar el comportamiento del coeficiente de reflexión con la frecuencia, que se presenta como un círculo en la gráfica de Smith:

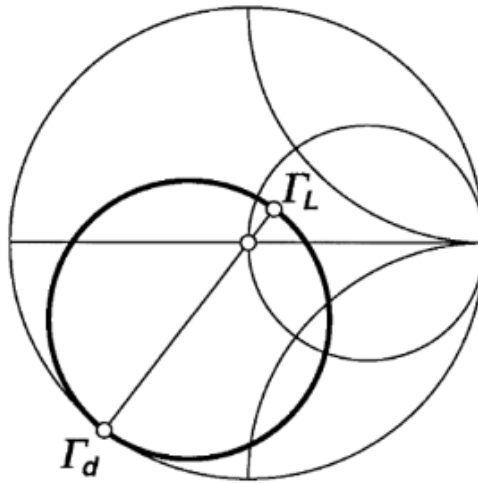


Figura 13. Coeficientes de reflexión del resonador

El valor del coeficiente de reflexión Γ_d (desintonizado) se obtiene cuando la frecuencia de operación está alejada de la frecuencia de resonancia de la cavidad. Sin embargo, cuando la frecuencia es igual a la frecuencia f_L (frecuencia de resonancia) se obtiene el valor del coeficiente de reflexión Γ_L . El diámetro que presenta el círculo Q es la distancia que existe entre ambos coeficientes de reflexión. Una vez obtenido dicho diámetro, se puede obtener el factor de acoplo como se muestra a continuación:

$$k = \frac{1}{\frac{2}{d} - 1} \quad (11)$$

donde:

- K: factor de acoplo
- d: diámetro

Se distinguen tres situaciones en función del valor del factor de acoplo (y por tanto, del diámetro del círculo):

- Acoplo crítico: Esto ocurre cuando $k = 1$. Esto implica que se disipa la misma cantidad de potencia en el circuito externo como en el propio resonador. El tener un factor de acoplo crítico permite saber que el factor de calidad descargado es el doble que el factor de calidad cargado.
- Acoplo subcrítico: Esto ocurre cuando $k < 1$. Esto implica que la potencia que disipa el circuito externo es menor a la potencia disipada en la cavidad resonante, lo que significa que los valores del factor de calidad cargado y descargado tienen menos diferencia.
- Acoplo sobrecrítico: Este caso se da cuando $k > 1$. En este caso la potencia que disipa el circuito externo es mayor a la potencia disipada por la cavidad resonante, por tanto, la diferencia entre los valores del factor de calidad cargado y descargado es mayor.

2.2.4. R/Q

Como se ha comentado en la introducción al TFM, uno de los programas realiza el Bead-Pull en una cavidad RFQ [9]. La medida Bead-Pull consiste en utilizar una perla (Bead), un metal o un dieléctrico para realizar perturbaciones en el campo a lo largo del eje de la cavidad. Se realiza un cambio en la frecuencia que es proporcional al cambio en el valor del campo.

Estas medidas se realizan de esta manera porque Slater demostró que el cambio en la frecuencia resonante al introducir un objeto en el campo de la cavidad es proporcional al cambio relativo en la energía almacenada. En la figura 14 se puede apreciar cómo se realiza la perturbación en un campo uniforme utilizando una perla dieléctrica:

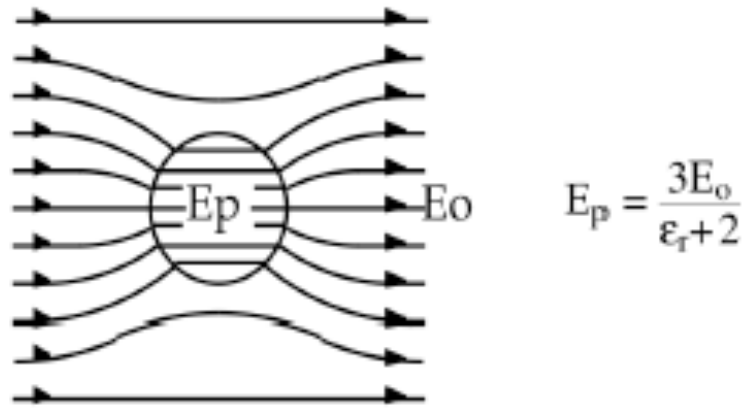


Figura 14. Perturbación en un campo uniforme utilizando una perla dieléctrica

En el caso de una pequeña esfera no conductora, de radio r , donde el campo no perturbado puede considerarse uniforme en una región más grande que la perla. Esto se puede demostrar en la ecuación a continuación [11]:

$$\frac{\Delta\omega}{\omega} = \frac{\Delta U}{U} = -\frac{\omega\pi r^3}{PQ} \left[\epsilon_0 \frac{\epsilon_r - 1}{\epsilon_r + 2} E_0^2 + \mu_0 \frac{\mu_r - 1}{\mu_r + 2} H_0^2 \right] \quad (12)$$

Z

Para una perla dieléctrica ($\mu_r = 1$) la expresión puede reducirse a la siguiente expresión:

$$\frac{\Delta\omega}{\omega} = -\frac{\pi r^3}{U} \left[\epsilon_0 \frac{\epsilon_r - 1}{\epsilon_r + 2} E_0^2 \right] \quad (13)$$

En el caso de una perla de metal ($\mu_r \rightarrow 0$ y $\epsilon_r \rightarrow \infty$) la expresión (11) se puede reducir de la siguiente manera:

$$\frac{\Delta\omega}{\omega} = -\frac{\pi r^3}{U} \left[\epsilon_0 E_0^2 + \mu_0 \frac{\mu_0}{2} H_0^2 \right] \quad (14)$$

Una perla metálica puede utilizarse para medir el campo eléctrico si se sabe que el campo magnético es cero (por ejemplo, en el eje de un modo monopolar), y da un cambio de frecuencia mayor que los materiales dieléctricos comunes como el teflón ($\epsilon_r = 2.08$) o la Alúmina ($\epsilon_r = 9.3$).

Las perlas que se pueden utilizar pueden presentar diferentes formas, como agujas o discos, para mejorar la perturbación y dar selectividad direccional. La mejora o el “factor de forma” puede calcularse para elipsoides o también se puede calibrar en un campo conocido.

Para poder calcular la resistencia transversal RT^2 o el factor R/Q [11] hay que realizar un mapeado de la distribución longitudinal del campo E_z e integrar la impedancia de derivación de la cavidad:

$$RT^2 = \frac{(VT)^2}{2P} = \frac{\left[\int E_z(Z) e^{jw \frac{Z}{v}} dZ \right]^2}{2P} \quad (15) \quad \text{donde } E^2 = -\frac{\Delta w P Q (\epsilon_r + 2)}{w^2 \pi r^3 \epsilon_0 (\epsilon_r - 1)}$$

Por tanto:

$$RT^2 = -\frac{Q(\epsilon_r + 2)}{\omega \pi r^3 \epsilon_0 (\epsilon_r - 1)} \frac{\left[\int \sqrt{\frac{\Delta \omega}{\omega}}(z) \left(\cos \frac{\omega z}{c} + j \sin \frac{\omega z}{c} \right) dz \right]^2}{2} \quad (16)$$

En el caso de que la cavidad sea simétrica en Z y en el centro $t=0$ en $Z=0$, entonces se cumple:

$$\frac{RT^2}{Q} = -\frac{Q(\epsilon_r + 2)}{\omega \pi r^3 \epsilon_0 (\epsilon_r - 1)} \left[\int \sqrt{\frac{\Delta \omega}{\omega}}(z) \left(\cos \frac{\omega z}{c} + j \sin \frac{\omega z}{c} \right) dz \right]^2 \quad (17)$$

Si se analiza la ecuación 16, se puede concluir que R/Q depende únicamente de la forma geométrica que presenta la cavidad. Algunos ingenieros eléctricos utilizan la definición de R/Q con un factor adicional de 2 en el denominador (definición de circuitos).

En el caso de querer calcular la impedancia de la derivada de la cavidad (R) habría que multiplicar por Q la ecuación 17, quedando la siguiente ecuación:

$$R = \frac{1}{\pi r^3 \epsilon_0} \frac{1}{2\omega} \left[\left(\int Q \frac{\Delta \omega}{\omega}(x, y, z) \right)^{1/2} dz \right]^2 \quad (18)$$

Sin embargo, si la perla es demasiado pequeña o el campo eléctrico de la cavidad es pequeño entonces la frecuencia de resonancia puede cambiar y puede complicar la medida. Una medida más sensible podría ser medir el cambio de fase del S_{21} de frecuencia de resonancia “no perturbada” cuando se introduce la perla en el interior de la cavidad. La impedancia de la cavidad viene definida en la fórmula a continuación [11]:

$$Z = R e^{j\phi} \cos \phi \quad \text{donde} \quad \tan \phi = Q \left(\frac{\omega_0}{\omega} - \frac{\omega}{\omega_0} \right) \quad (19)$$

$$\text{si } \omega = \omega_0 \left(1 + \frac{\Delta \omega}{\omega_0} \right) \quad \text{donde} \quad \frac{\Delta \omega}{\omega_0} \ll 1 \quad (20)$$

Entonces se cumple que:

$$Q \frac{\Delta\omega}{\omega_0} \cong \frac{1}{2} \tan\phi \quad (21)$$

Para valores pequeños del ángulo phi, se puede aproximar la expresión (21) por:

$$\phi \cong 2Q \frac{\Delta\omega}{\omega_0} \quad (21)$$

De forma que, en condiciones de pequeñas perturbaciones, el desfase está relacionado linealmente con la variación relativa de frecuencia y el factor de calidad. Lo cual es una razón más para tratar de obtener un valor lo más exacto posible de Q.

2.3. Medida del Factor de calidad

Para medir el factor Q de un resonador de microondas existen varios métodos. En un primer momento Ginzton [13] ideó la medida del factor de calidad en la que el coeficiente de reflexión se ajustaba gráficamente a un círculo, es decir, desarrollando un primer método de los tres puntos. La característica principal del procedimiento de Ginzton es que la cantidad de acoplo está determinada por el mismo experimento, de modo que se puede eliminar la influencia que ejerce el acoplo y calcular la propiedad inherente del factor de calidad (Q) del propio resonador descargado.

A continuación, se van a describir los procedimientos de medida del factor de calidad utilizando los métodos que se han implementado en el programa:

2.3.1. Métodos de los 3 puntos

En primer lugar, se va a describir el procedimiento a seguir para calcular la Q en reflexión [13]. Primero hay que identificar el modo resonante. Para ello se conecta el puerto 1 del VNA (analizador de redes vectorial) al acoplador seguido de la cavidad y se mide la respuesta de reflexión, S_{11} , con la frecuencia. Se presenta S_{11} en la pantalla del VNA en formato logarítmico (dB) y se sitúa un marcador en la frecuencia de resonancia, que coincidirá con el mínimo del S_{11} . El siguiente paso es visualizar el S_{11} en el diagrama de Smith, lo que permite que se vea un círculo. Cuanto más grande sea éste, mayor será el factor de acoplo. Se distinguen tres situaciones en función del valor del factor de acoplo, β (y por tanto, del tamaño del círculo) [13]:

- Acoplo subcrítico ($0 < \beta < 1$): El círculo de S_{11} presenta un diámetro menor que uno, y no rodea al centro del gráfico de Smith.
- Acoplo crítico ($\beta = 1$): El círculo toca el centro del gráfico a la frecuencia de resonancia
- Acoplo sobrecrítico ($\beta > 1$): El centro del gráfico queda dentro del círculo.

En el caso de que sea posible, se intenta ajustar el factor de acoplo en torno a 1, de forma que la reflexión a la frecuencia de resonancia sea mínima.

Se aplica un desfase para que el círculo de S_{11} se sitúe sobre el eje real del gráfico de Smith, de forma que lejos de la resonancia, el S_{11} quede localizado en la ubicación del corto circuito.

La Q cargada (Q_L) se puede determinar a partir de la frecuencia de resonancia, f_0 , y las frecuencias f_1 y f_2 , donde la parte imaginaria de S_{11} es mínima y máxima, respectivamente. A continuación, se muestra la fórmula de la Q cargada basada en el método de los tres puntos [13]:

$$Q_L = \frac{f_0}{f_2 - f_1} \quad (22)$$

El valor de Q descargado (Q_u) se puede determinar utilizando la siguiente relación:

$$Q_u = Q_L(1 + \beta) \quad (23)$$

En el caso particular de que haya acoplo crítico ($\beta = 1$), el valor de Q descargado quedaría definida de la siguiente manera:

$$Q_u = 2Q_L \quad (24)$$

Existe asimismo una forma de determinar el valor de Q descargado en reflexión. Para ello se utiliza las frecuencias f_3 y f_4 , donde se igualan las partes imaginarias y reales de la impedancia de entrada (esto es, donde $\text{Im}(z) = \text{Re}(Z)$ y $\text{Im}(Z) = -\text{Re}(Z)$ respectivamente). A continuación, se muestra la fórmula que permite calcular dicha Q:

$$Q_u = \frac{f_0}{f_4 - f_3} \quad (25)$$

Una vez realizado el cálculo de Q en reflexión, se va a realizar el cálculo en transmisión. De la misma manera que se ha hecho en la medida de reflexión, se conecta el analizador de redes al resonador y se representa el S_{11} , S_{22} y el S_{21} . Una vez conectado, es conveniente visualizar la respuesta en transmisión, S_{21} , en formato logarítmico y las respuestas en reflexión, S_{11} y S_{22} , en formato carta de Smith, lo cual permite ver los círculos correspondientes a la Q.

La Q cargada (Q_L) [13], viene determinada por las frecuencias f_1 y f_2 , las cuales determinan el ancho de banda 3 dB del S_{21} . A continuación, se muestra la fórmula correspondiente a la Q cargada en transmisión:

$$Q_L = \frac{f_0}{f_2 - f_1} \quad (26)$$

En la figura siguiente se muestra el cálculo del factor de calidad utilizando este método de los tres puntos, es decir, utilizando la expresión (22). Los marcadores que se han utilizado para realizar el cálculo en este caso son el marcador 1 para la frecuencia central ($f_0 = \text{marcador 1} = 361 \text{ MHz}$), y los marcadores 2 y 3 para las frecuencias laterales ($f_1 = \text{marcador 2}$ y $f_2 = \text{marcador 3}$)

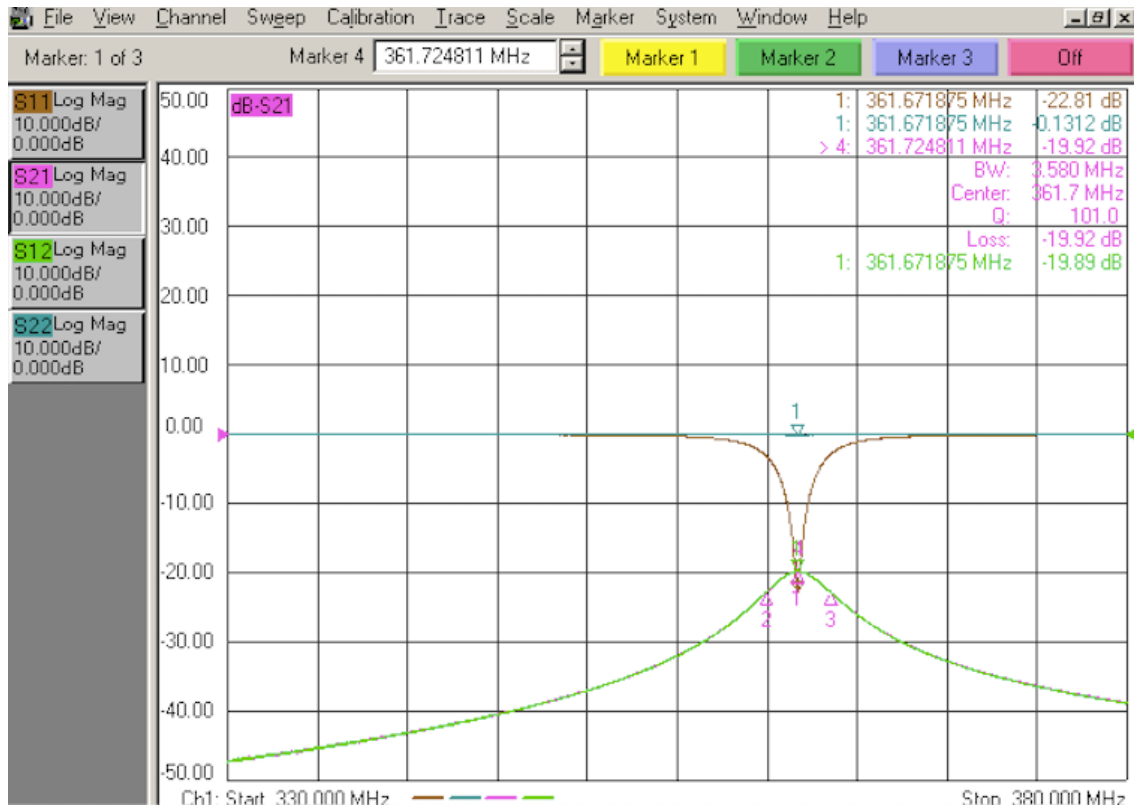


Figura 15. Cálculo de la Q utilizando el método de los tres puntos.

Se pueden obtener los factores de acoplo fácilmente visualizando los marcadores del VNA en la frecuencia f_0 en formato SWR (ROE). En el caso de que estuviese infra acoplado $K_m = 1/\text{SWR}$. En caso de acoplo crítico o sobreacoplo, $K_m = \text{SWR}$ (nótese que $K = \beta$).

Los valores de los acoplos medidos en el modo de transmisión difieren de los valores reales debido a la influencia cruzada entre ambos acoplos. Los acoplos reales [13] se pueden calcular de la siguiente manera:

$$k_1 = k_{m1} \left(\frac{1 + k_{m2}}{1 - k_{m1}k_{m2}} \right) \quad (27)$$

$$k_2 = k_{m2} \left(\frac{1 + k_{m1}}{1 - k_{m1}k_{m2}} \right) \quad (28)$$

Donde $K_{m1,2}$ son los factores de acoplo aparentes medidos en el modo de transmisión mientras que el $K_{1,2}$ son los acoplos reales, como se puede medir en el modo de reflexión. Se puede demostrar que $K_{m1,2}$ son siempre menores que $K_{1,2}$ y que cuanto más fuerte es el acoplo en un puerto, mayor es el efecto en el otro puerto.

El valor de Q descargado (Q_u) puede determinarse usando la ecuación que se muestra a continuación, que es una generalización de la ecuación (23):

$$Q_u = Q_L(1 + k_1 + k_2) \quad (29)$$

Sólo en el caso de acoplos iguales en ambos puertos, el valor de Q descargado (Q_u) puede determinarse a partir de la Q cargada (Q_L) obtenida mediante la expresión (22) y las pérdidas de inserción en la resonancia, sin necesidad de atender a la medida en reflexión o los factores de acoplo. A continuación, se muestra la fórmula que permite realizar este cálculo:

$$Q_u = \frac{Q_L}{1 - S_{21}(f_0)} \quad (30)$$

Esto se cumple siempre que $k_1 = k_2$, y donde $S_{21}(f_0)$ son las pérdidas en magnitud lineal:

$$S_{21}(f_0) = 10^{\left(\frac{dB_{S_{21}}(f_0)}{20}\right)} \quad (31)$$

Los métodos descritos en este punto, como se ha comentado al inicio, corresponden a los métodos de los tres puntos. Estos métodos permiten dar una estimación rápida del factor de calidad y los coeficientes de acoplo usando una calculadora o una hoja de cálculo, pero no dejan de ser una aproximación que se ve sensiblemente afectada por errores de medición aleatorios, que son particularmente severos en los casos de acoplo débil o factor de calidad alto.

De manera alternativa, el factor de calidad tanto cargado como descargado y los factores de acoplo ($K = \beta$), pueden calcularse utilizando métodos “sobredeterminados” aplicando un procedimiento de ajuste de curvas.

Este ajuste de curva (“curve fitting”) es el procedimiento que plantean utilizar tanto D. Kajfez como posteriormente K. Leong.

2.3.2. Método de Kajfez

En la actualidad se utiliza ampliamente el analizador vectorial de redes, éste genera datos de coeficientes de reflexión de alta precisión que son muy adecuados para el procesamiento de datos numéricos por una computadora, esto ha supuesto el remplazo del procedimiento gráfico de Ginzton.

El procedimiento numérico que va a describirse a continuación es una aplicación del ajuste de datos para una transformación fraccional lineal en el plano complejo. La aplicación del procedimiento está destinada a valores altos del factor Q, típicamente de varios de miles, como los que se pueden encontrar en las cavidades resonantes para aceleradores de partículas. En este método no se ignoran las pérdidas, por pequeñas que sean, en el circuito de acoplo. Presenta un modelo de corrección con una resistencia en serie que permite representar las pérdidas de acoplo.

Para medir el tipo de reflexión de una cavidad resonante de Q alto se puede representar mediante el circuito de la figura 16.

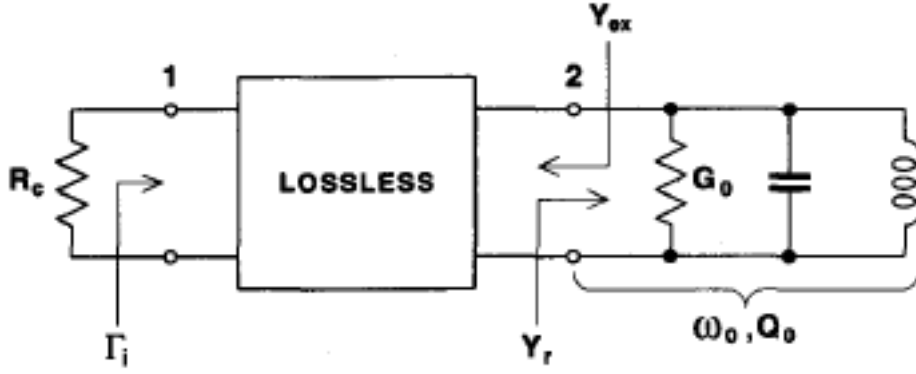


Figura 16. Circuito resonador acoplado a un circuito externo de dos puertos sin pérdidas [14].

Este circuito [14] es un circuito resonante paralelo de frecuencia de resonancia (pulsación, $\omega_0 = 2\pi f_0$) y factor Q (Q_0) que representa el modo resonante en la cavidad sin carga. La red de dos puertos sin pérdidas representa el mecanismo de acoplo, que en este momento se supone que es recíproco y sin pérdidas. La red de acoplo entre el circuito resonador y el circuito externo está conectada a un dispositivo de medición, que se supone que tiene una resistencia interna R_c . Se puede suponer que las medidas se realizan en una banda estrecha de frecuencias y que las propiedades del acoplo son no varían en dicho rango de frecuencias.

Existen dos maneras diferentes de analizar el circuito de la figura 16. Si el observador está mirando el puerto 2 (posición de referencia del resonador descargado), necesitará expresar el comportamiento de la admitancia (Y_r) del resonador descargado, la cual está cargada por la admitancia del circuito externo (Y_{ex}). En el caso de realizar la observación desde el puerto 1 (posición de referencia de la medida del coeficiente de reflexión), necesita expresar el comportamiento del coeficiente de reflexión de entrada Γ_i en función de la frecuencia. Ambas observaciones deben conducir a una definición unificada de las características cuantitativas del circuito.

La admitancia que presenta el resonador cuando esté descargado puede calcularse según la fórmula que se muestra a continuación [14]:

$$Y_r = G_0 \left(1 + jQ_0 2 \frac{\omega - \omega_0}{\omega_0} \right) \quad (32)$$

La admitancia que presenta el circuito externo es:

$$Y_{ex} = \frac{A + CR_c}{B + DR_c} = G_{ex} + jB_{ex} \quad (33)$$

En la fórmula (33), desde la A hasta la D son constantes pertenecientes a la matriz de transmisión [ABCD]. En el caso de redes de dos puertos recíprocas y sin pérdidas, A y D son constantes reales y B y C son imaginarias. Además, se cumple que $AD - BC = 1$. La parte real de la admitancia externa se puede obtener de la siguiente manera:

$$G_{ex} = \frac{R_c}{|B + DR_c|^2} \quad (34)$$

La conductancia del circuito externo es la que se encarga de cargar el circuito resonante, de manera que el factor de calidad cargado se convierte en Q_L de la siguiente manera:

$$Y_L = Y_{ex} + Y_r = (G_0 + G_{ex}) \left(1 + jQ_L 2 \frac{\omega - \omega_L}{\omega_L} \right) \quad (35)$$

La susceptancia externa mueve la frecuencia de resonancia del circuito, por esta razón la frecuencia resonante del circuito pasa a valer ω_L :

$$\omega_L = \omega_0 \left(1 - \frac{B_{ex}}{2Q_0G_0} \right) \quad (36)$$

Todos los elementos del circuito de dos puertos tienen un voltaje común, el coeficiente de acoplo puede definirse como la relación entre la potencia disipada y la potencia disipada en el circuito externo, es decir:

$$K = \frac{G_{ex}}{G_0} \quad (37)$$

En este punto se va a analizar lo que ocurre en el puerto 1. La impedancia de entrada se expresa en términos de parámetros transmisión de la función fraccional lineal:

$$Z_i = \frac{A + BY_r}{B + DY_r} \quad (38)$$

Por tanto, el coeficiente de reflexión en la entrada se puede definir de la siguiente manera:

$$\Gamma_i = \frac{Z_i - R_c}{Z_i + R_c} \quad (39)$$

En términos de admitancia descargada (Y_r) el coeficiente de reflexión es también una función fraccional lineal. A continuación, se muestra el coeficiente de reflexión en función de la admitancia descargada:

$$\Gamma_i = \frac{KY_r - L}{MY_r + 1} = \frac{K}{M} + \frac{L - \frac{K}{M}}{M} \frac{1}{Y_r + \frac{1}{M}} \quad (40)$$

Las constantes K y M se pueden expresar en términos de los parámetros de la matriz de transmisión. El término $1/M$ que aparece en el denominador de la fórmula (39), tiene el mismo valor que presenta la admitancia del circuito externo (Y_{ex}) (fórmula (32)):

$$\frac{1}{M} = \frac{A + CR_c}{B + DR_c} = Y_{ex} \quad (41)$$

La relación anterior implica que el denominador del coeficiente de reflexión (Γ_i) tiene la misma función de frecuencia que la admitancia Y_L . Además, la relación K/M es igual al coeficiente de reflexión Γ_s . A continuación, se muestra la relación:

$$\Gamma_s = \frac{K}{M} = \frac{B - DR_c}{B + DR_c} \quad (42)$$

En vista de la simplificación realizada al comienzo de este punto, las constantes de la “A” a la “D” no dependen de la frecuencia. En consecuencia, el coeficiente de reflexión de entrada es la siguiente ecuación en función de la frecuencia:

$$\Gamma_i = \Gamma_s + \frac{de^{-j2\delta}}{1+jQ_L2\frac{\omega-\omega_L}{\omega_L}} \quad (43)$$

El coeficiente de reflexión describe un círculo. De la ecuación (43) se puede deducir que el diámetro del factor Q es:

$$d = \left| \frac{L - \frac{K}{M}}{M} \right| \frac{1}{G_0 + G_{ex}} \quad (44)$$

En la siguiente figura (figura 17) se puede observar el coeficiente de reflexión a la entrada del circuito acoplado sin pérdidas. Además, se puede observar la frecuencia de resonancia inicial y la ω_L que pasa a ser la frecuencia de resonancia una vez que susceptancia desplaza la frecuencia inicial.

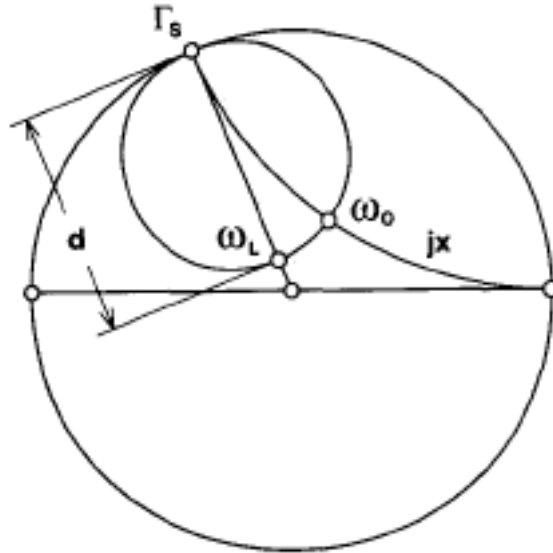


Figura 17. Coeficiente de reflexión a la entrada en función de la frecuencia, en el circuito sin pérdidas acoplado.

Si hacemos un análisis más profundo de la figura anterior, se puede observar que el círculo Q toca con la circunferencia de la carta de Smith en el punto Γ_s . El diámetro del círculo Q se puede también calcular en función de los términos A, B, C y D. A continuación, se muestra como calcular el diámetro del círculo Q en función de dichos círculos:

$$d = \frac{2R_c}{|B+DR_c|^2} \frac{1}{G_0 + G_{ex}} \quad (45)$$

El primer factor de la fórmula (45) se puede reorganizar como G_{ex} quedando de la siguiente manera el diámetro del círculo Q:

$$d = \frac{2G_{ex}}{G_0 + G_{ex}} \text{ si } k = \frac{G_{ex}}{G_0} \text{ entonces } d = \frac{2k}{1+k} \text{ donde } k = \frac{d}{2+d} \quad (46)$$

Como se puede observar en (39) la frecuencia de resonancia cargada (ω_L) es la frecuencia a la que el valor absoluto $|\Gamma - \Gamma_s|$ es máximo. Por tanto, esto implica que la ecuación (40) puede reinscribirse de la siguiente manera:

$$\Gamma_i = \frac{jQ_L\Gamma_s 2^{\frac{\omega - \omega_L}{\omega_0} + \Gamma_s + de^{-j2\delta}}}{jQ_L 2^{\frac{\omega - \omega_L}{\omega_0} + 1}} \quad (47)$$

Si la ecuación (49) se plantea como una transformación lineal fraccional (Mobious transformación) queda como se muestra a continuación:

$$\Gamma_i = \frac{a_1 t + a_2}{a_3 t + 1}, \text{ donde } t \text{ es la variable frecuencia normalizada, } t = 2 \frac{\omega - \omega_L}{\omega_0} \quad (48)$$

Hay N puntos medidos de Γ_i , uno para cada frecuencia. Los datos medidos se pueden ajustar utilizando la ecuación (48), dando como resultado tres números complejos (a_1 , a_2 y a_3). El procedimiento de ajuste de curva (curve fitting) consiste básicamente en escribir la ecuación (47) como combinación lineal desde a_1 hasta a_3 . A continuación, se muestra como quedaría dicha combinación lineal:

$$a_1 t_n + a_2 - a_3 t_n \Gamma_{i,n} = \Gamma_{i,n} \dots n = 1, 2, 3, \dots, N \quad (49)$$

El resultado de la combinación lineal es un sistema indeterminado de N ecuaciones y 3 incógnitas. El sistema se resuelve usando mínimos cuadrados, su solución proporciona las tres incógnitas (a_1 hasta a_3) y sus varianzas $\sigma(a_i)$ para $i = 1, 2, 3$. Las varianzas de los coeficientes a_i son útiles para obtener las estimaciones de error de las cantidades medidas.

Los resultados anteriores son muy sensibles al valor correcto del ω_L , por esta razón, el ajuste se realiza dos veces. La primera vez, la frecuencia que se mide es la más cercana al centro de la carta de Smith y toma el valor de ω_L . Una vez que se obtiene el radio y el centro del círculo, se evalúa un mejor valor para ω_L y el procedimiento de ajuste de curva se realiza entonces por segunda vez para este nuevo valor de ω_L .

Una vez resuelto el sistema, también se puede conocer el valor de la Q cargada, que de acuerdo con la ecuación (47) es la parte imaginaria de a_3 .

$$Q_L = \text{Im}(a_3) \quad (50)$$

El diámetro del círculo Q también puede calcularse en función de a_1 , a_2 y a_3 . Por tanto, una vez resuelto el sistema también se puede obtener el diámetro del círculo. A continuación, se muestra cómo se calcularía el diámetro en función de dichas incógnitas:

$$d = \left| a_2 - \frac{a_1}{a_3} \right| \quad (51)$$

Una vez que se ha calculado el coeficiente de acoplo se puede calcular el valor de Q descargado de la siguiente manera:

$$Q_o = Q_L(a + k) \quad (52)$$

Un caso especial del acoplo de dos puertos es una reactancia en serie simple (X_s). En este caso, el valor de R_s se establece en cero y por tanto el coeficiente de acoplo que se había calculado en la ecuación (36) pasa a ser:

$$k = \frac{G_{ex}}{G_0} = \frac{G_C}{G_0(1+x^2)} \quad (53) \text{ normalizando reactancia } x = \frac{X_S}{R_C} \quad (54)$$

La forma en la que Ginzton definió el coeficiente de acoplo se muestra a continuación:

$$\beta = \frac{G_C}{G_0} \quad (55)$$

De manera que para el circuito equivalente (figura 18) y una $R_s = 0$, la relación del coeficiente de acoplo de Ginzton (β) con el coeficiente de acoplo k de la ecuación (36) se convierte en:

$$k = \frac{\beta}{1+x^2} \quad (56)$$

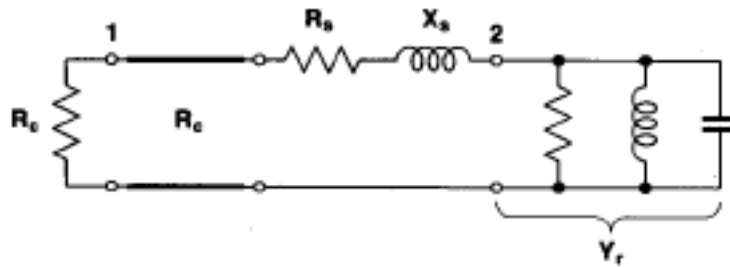


Figura 18. Circuito equivalente donde R_s vale cero

En algunas situaciones, el resonador dieléctrico, montado junto a una línea de transmisión microstrip, el valor de X es considerablemente más pequeño que la unidad de modo que la diferencia en las definiciones anteriores no se nota. En otras situaciones, el acoplo en bucle el valor medido de X puede ser mayor que la unidad, lo que implica un valor reducido del diámetro del círculo Q , de manera que las dos definiciones producen valores muy diferentes del coeficiente de acoplo.

Kajfez además plantea una alternativa para corregir las pérdidas del circuito de acoplo. En una disposición de acoplo típica entre el circuito externo y el resonador de microondas, el conector coaxial termina en un lazo como se muestra en la figura 19. Este lazo corresponde con X_s en la figura 18. Si el conductor del lazo tiene una resistencia apreciable, el circuito equivalente se puede corregir sumando la resistencia R_s en serie con la reactancia X_s , como se muestra en el circuito equivalente de la figura 18.

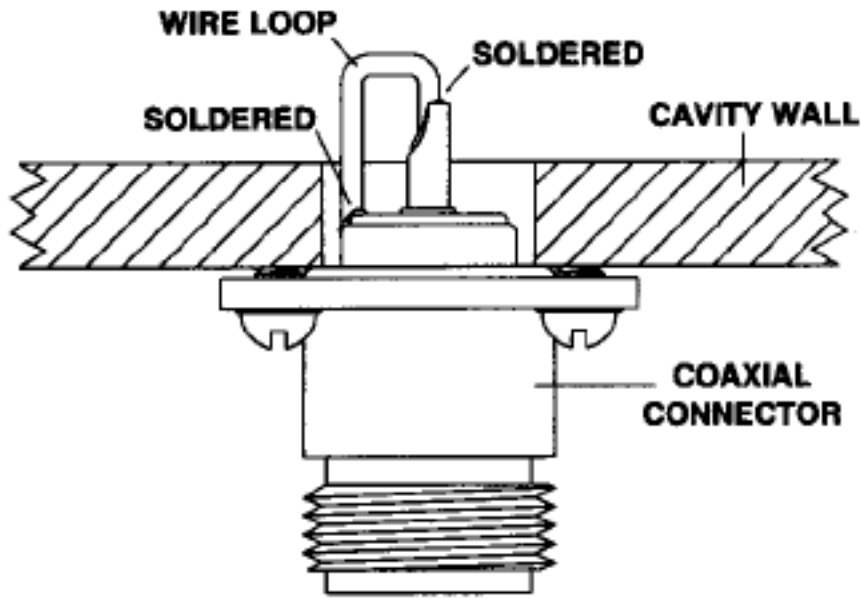


Figura 19. Típico lazo de acoplo

El coeficiente de reflexión de entrada del circuito se puede seguir expresando con la ecuación (43), pero ahora el valor absoluto del coeficiente de reflexión desintonizado tendrá un valor menor a la unidad. Una interpretación gráfica de este hecho es que el círculo Q nunca toca la circunferencia unidad de la carta de Smith.

A continuación, se muestra la ecuación que permite calcular el acoplo cuando hay pérdidas:

$$k_S = \frac{R_S}{R_C} \quad (57)$$

Existen varios parámetros de interés que se calculan de diferente manera ahora que hay pérdidas en el acoplo. A continuación, se muestran dichos parámetros:

- Coeficiente de reflexión Γ_S : $\Gamma_S = \frac{a_1}{a_3}$ (58)
- Diámetro del círculo Q: $d = 2 \left| \frac{a_2 a_3 - a_1}{a_3 - a_S^*} \right|$ (59)
- Diámetro d_2 del círculo que toca la circunferencia de Smith: $d_2 = 2 \frac{1 - |\Gamma_S|^2}{1 - |\Gamma_S| \cos \Phi}$ (60)
- Acoplo con pérdidas: $k_S = \frac{2}{d_2} - 1$ (61)

Finalmente, el coeficiente de acoplo, cuando hay pérdidas por acoplo, se puede calcular acorde con la ecuación (37):

$$k = \frac{d}{d_2 - d} \quad (62)$$

Concluyendo, esta teoría permite medir el factor de calidad Q en resonadores con una alta Q utilizando un ajuste de curvas. Este procedimiento de transformación lineal en un plano complejo es un procedimiento numérico que reemplaza el procedimiento tradicional de

ajuste de datos. El procedimiento acepta los coeficientes de reflexión medidos en un orden arbitrarios y no requiere ningún acondicionamiento previo de datos.

2.3.3. Método de K. Leong

Las medidas del Q descargado (Q_0) [15] tienen un papel importante en la evolución del desempeño de los sistemas y propiedades de los materiales. Los diferentes valores del factor Q son especialmente útiles en las mediciones de selectividad de sistemas, así como la resistencia superficial de materiales conductores y superconductores y la permitividad compleja de materiales dieléctricos. Por definición el factor de calidad es una relación entre la energía máxima (W_{max}) y almacenada dentro de un sistema de energía disipada W_α .

$$Q = \frac{2\pi W_{max}}{W_\alpha} \quad (63)$$

La mayoría de las técnicas para medir el factor Q están desarrollados en base a modelos ideales de sistemas resonadores. Por eso, cualquier discrepancia entre el resonador ideal y el sistema práctico de medición del resonador y una inadecuada calibración puede producir error en la medida del factor de calidad. Recientemente, se ha mostrado un fuerte interés en la caracterización precisa de microondas de nuevos materiales a temperaturas de criogenia para aplicaciones de comunicaciones inalámbricas. La mayoría de los sistemas de medición de criogenia contiene cables y adaptadores, que no se pueden calibrar fácilmente, por tanto, se ha creado un nuevo y preciso método de medición del factor de calidad, que tiene en cuenta los efectos parásitos de medición. Dicho método se va a describir en este punto.

Para poder desarrollar un método factible, preciso y útil para determinar el factor Q de resonadores dieléctricos, se han tenido que realizar ciertas suposiciones:

- El método es aplicable a resonadores que trabajan en modo de transmisión.
- La ecuación completa para calcular el factor de calidad descargado es la siguiente:

$$Q_0 = Q_L(1 + \beta_1 + \beta_2) \quad (64)$$

Implicando lo siguiente:

- Se necesitará una nueva ecuación para relacionar el factor Q cargado con el parámetro S_{11} , alrededor de la resonancia de un modelo de circuito del resonador dieléctrico.
- Se necesita una nueva ecuación para relacionar los coeficientes de acoplo (β_1 y β_2) con los parámetros de reflexión (S_{11} y S_{22}) para el modo de transmisión del resonador.
- Derivando las ecuaciones del factor de calidad cargado (Q_L), coeficiente de acoplo y la frecuencia en función del S_{21} , es la mejor relación para crear un modelo de medición del sistema lo más real posible.

- Los parásitos de los parámetros son importantes a la hora de realizar una medida con el resonador real, pero no están incluidos en el modelo de circuito del sistema. Dichos parásitos serán contabilizados en el procedimiento de ajuste de curvas (curve-fitting) aplicándolo a la medida de los parámetros S del sistema.

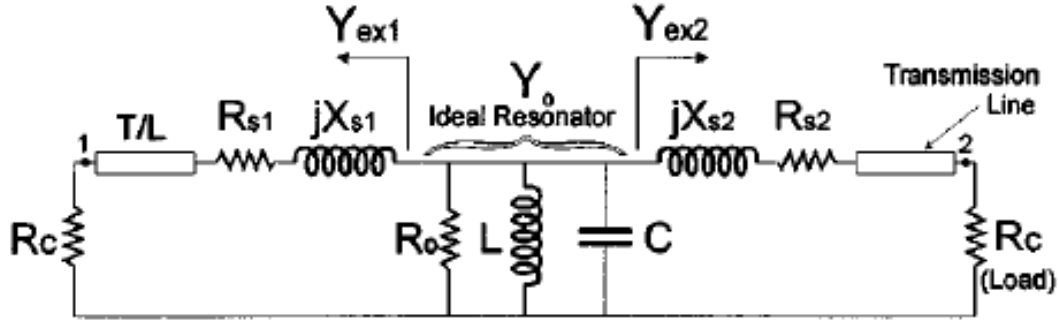


Figura 20. Modelo del circuito en transmisión del sistema de un resonador.

Para obtener las ecuaciones que relacionan los parámetros S con el factor Q cargado (Q_L), el coeficiente de acoplo y la frecuencia de resonancia se utiliza el modelo que está representado en la Figura 20.

Los tres componentes R_0 , L y C , correspondientes a la admitancia Y_0 , representan el resonador dieléctrico ideal. Estos elementos son quienes determinan el factor Q descargado (Q_0) del sistema. La admitancia se puede calcular con la siguiente aproximación:

$$Y_0 = \frac{1}{R_0} \left\{ 1 + j2Q_L \frac{\omega - \omega_0}{\omega_0} \right\} \quad (65)$$

Donde Q_0 es el factor de calidad descargado del resonador y ω_0 es la frecuencia de resonancia, en radianes, del resonador ideal y se puede calcular de la siguiente manera:

$$\omega_0 = \frac{1}{\sqrt{LC}} \quad (66)$$

Las admitancias Y_{ex1} y Y_{ex2} , son admitancias externas vistas en cada uno de los puertos del resonador ideal, como se puede observar en la figura 20. Se pueden calcular con la siguiente ecuación:

$$Y_{exp} = \frac{1}{R_{SP} + R_C + jX_{SP}} = G_{exp} + jB_{exp} \quad (67)$$

Donde la P de 1 o 2 denota el puerto de entrada y salida del resonador respectivamente. Las resistencias R_{s1} y R_{s2} representan las pérdidas por las estructuras de acoplo, conectores, pérdidas en las líneas de transmisión que conectan los dos puertos del resonador; reactancias de las estructuras del acoplo se pueden modelar como X_{s1} y X_{s2} . La resistencia R_C representa una resistencia interna de la fuente de microondas, es una carga e impedancia característica de las líneas de transmisión en ambos puertos.

El diagrama del circuito de la figura 20 se puede considerar como tres conexiones en cascada de tres redes y puede describirse usando una matriz de transmisión [ABCD]:

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} 1 & R_{S1} + jX_{S1} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ Y_0 & 1 \end{bmatrix} \begin{bmatrix} 1 & R_{S2} + jX_{S2} \\ 0 & 1 \end{bmatrix} \quad (68)$$

Las líneas de transmisión se han considerado ideales y por esa razón no están incluidas en la matriz (67). Las pérdidas y los retrasos se contabilizarán más adelante.

Los parámetros S del sistema total del resonador pueden obtenerse a través de los parámetros “ABCD” utilizando las siguientes ecuaciones:

$$S_{21} = \frac{2R_C}{AR_C + B + CR_C^2 + DR_C} \quad (69)$$

$$S_{11} = \frac{(A-D)R_C + B - CR_C^2}{(A+D)R_C + B + CR_C^2} \quad (70)$$

$$S_{22} = \frac{(D-A)R_C + B - CR_C^2}{(A+D)R_C + B + CR_C^2} \quad (71)$$

Sustituyendo la ecuación (68) en la ecuación (69) queda la siguiente expresión para coeficiente de transmisión S_{21} :

$$S_{21} = \frac{2R_C Y_{ex1} Y_{ex2}}{Y_{ex1} + Y_{ex2} + Y_0} \quad (72)$$

El denominador de la expresión (72) es igual a la admitancia total (Y_L) donde:

$$Y_L = Y_0 + Y_{ex1} + Y_{ex2} = G_0 + G_{ex1} + G_{ex2} + j2Q_0 G_0 \frac{\omega - \omega_0}{\omega_0} + j(B_{ex1} + B_{ex2}) \quad (73)$$

La frecuencia de resonancia ω_L en el resonador cargado se puede calcular a partir de la expresión (73), asumiendo que la parte imaginaria de la admitancia total (Y_L) es cero.

$$\omega_L = \omega_0 \left[1 - \frac{B_{ex1} + B_{ex2}}{2Q_0 G_0} \right] \quad (74)$$

Si se parte de las ecuaciones (69) hasta la (74) se puede observar que existe una dependencia entre el coeficiente de transmisión complejo del S_{21} y los modos resonadores en transmisión. Los modos se pueden expresar mediante una nueva ecuación donde se relaciona el S_{21} con el factor Q cargado (Q_L), el coeficiente de acoplo y la frecuencia. A continuación, se muestra dicha relación:

$$S_{21}(\omega) = \frac{2R_C Y_{ex1} Y_{ex2}}{G_0(1+\beta_1+\beta_2) \left[1 + j2Q_L \frac{(\omega - \omega_L)}{\omega_0} \right]} \approx \frac{2R_C Y_{ex1} Y_{ex2}}{G_0(1+\beta_1+\beta_2) \left[1 + j2Q_L \frac{(\omega - \omega_L)}{\omega_0} \right]} \quad (75)$$

Donde β_1 y β_2 son los coeficientes de acoplo del resonador cargado. Los coeficientes de acoplo se pueden definir de la siguiente manera:

$$\beta_1 = \frac{G_{ex1}}{G_0} \quad \beta_2 = \frac{G_{ex2}}{G_0} \quad (76)$$

Para calcular el factor Q cargada (Q_L) del resonador se aplica la siguiente expresión:

$$Q_L = Q_0 \left(\frac{G_0}{G_0 + G_{ex1} + G_{ex2}} \right) \quad (77)$$

La aproximación de $\omega_0 \approx \omega_L$ puede considerarse aceptable para sistemas que presentan un factor de calidad alto. La ecuación (75) tiene una forma de lineal fraccional o bilineal. A continuación, se muestra la ecuación bilineal:

$$S_{21} = \frac{a_1 t + a_2}{a_3 t + 1} \quad (78)$$

Donde las constantes a_1 , a_2 y a_3 tienen el siguiente valor:

$$a_1 = 0 \quad a_2 = \frac{2R_C Y_{ex1} Y_{ex2}}{G_0(1 + \beta_1 + \beta_2)} \quad a_3 = jQ_L \quad (79)$$

Y “t” es una variable de la frecuencia normalizada que se define como:

$$t = 2 \left(\frac{\omega - \omega_L}{\omega_L} \right) \quad (80)$$

El parámetro a_3 , su parte imaginaria es igual al valor del factor de calidad cargado (Q_L), es decir:

$$Q_L = \text{Im}[a_3] \quad (81)$$

Los parámetros a_i se obtienen usando un ajuste de curva lineal fraccional (curve fitting) aplicado a la expresión (75). La relación que hay entre los parámetros de reflexión S_{11} y S_{22} y el factor de calidad cargado (Q_L) se obtiene sustituyendo las ecuaciones (68)-(70) en la expresión (71), dando como resultado:

$$S_{pp}(w) = \frac{jQ_L S_{ppd} 2 \frac{\omega - \omega_L}{\omega_L} + \left\{ S_{ppd} + \frac{2R_C Y_{exp}^2}{G_0(1 + \beta_1 + \beta_2)} \right\}}{jQ_L 2 \frac{\omega - \omega_L}{\omega_L} + 1} \quad (82)$$

En la ecuación anterior, el parámetro S_{ppd} representa el valor desplazado en frecuencia del coeficiente de reflexión en el puerto “p” (puerto 1 o 2) cuando la frecuencia está lejos de ω_L . De la misma manera que ocurre con el S_{21} , se puede expresar el S_{pp} como una fracción lineal de la siguiente manera:

$$S_{pp} = \frac{a_1 t + a_2}{a_3 t + 1} \quad (83)$$

Por tanto, la fórmula (82) también es adecuada para realizar un ajuste de curva lineal fraccional. Las constantes complejas de la expresión (83) tienen relación con puntos importantes del círculo Q: el radio a_1/a_3 representa el valor desplazado de la parte compleja del coeficiente de reflexión S_{ppd} , a_2 es el coeficiente de reflexión a la frecuencia de resonancia y la parte imaginaria de a_3 como se ha comentado antes es igual al valor de Q_L .

$$a_1 = jQ_L S_{ppd} \quad a_2 = S_{ppd} + \frac{2R_C Y_{exp}^2}{G_0(1+\beta_1+\beta_2)} \quad a_3 = jQ_L \quad (84)$$

La parte compleja de la constante a_3 es la misma solución para la transmisión, lo que significa que el modo resonante de transmisión la Q_L puede localizarse de la misma manera que en el modo reflexión. El diámetro del círculo de reflexión se puede obtener a partir de los puntos de resonancia y los puntos completamente desintonizados, los cuales están simétricamente opuestos los unos con los otros en el círculo de Q .

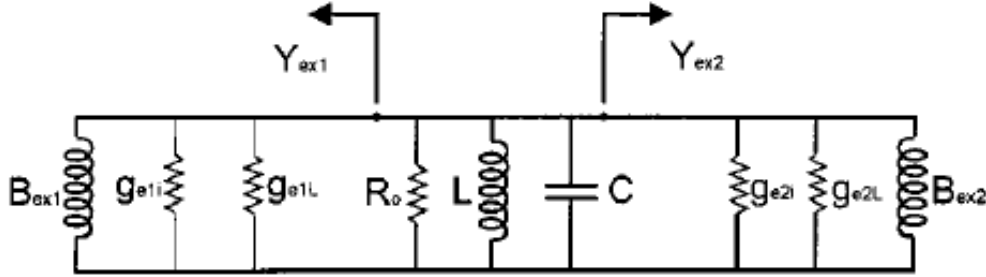


Figura 21. Circuito equivalente en transmisión del resonador con pérdidas en acoplo con pérdidas [15]

Los valores del coeficiente de acoplo de un resonador en transmisión no se pueden obtener directamente a partir de los valores de S_{11} y S_{22} medidos cerca de la frecuencia de resonancia. Para resolver el problema, se emplea una solución que divide los coeficientes de acoplo en dos partes: con pérdidas y sin pérdidas. Esta solución ya se ha aplicado con anterioridad en la expresión (75) para los resonadores en modo reflexión. En el caso en el que el acoplo presenta pérdidas se utiliza el modelo del circuito equivalente que aparece en la figura 21. El modelo de la figura 17 es similar al modelo en modo reflexión (un puerto) del resonador, pero en este caso con dos puertos. El coeficiente de acoplo en cada uno de los puertos del resonador se puede expresar como se muestra a continuación:

$$\beta_p = \frac{G_{exp}}{G_0} = \left(\frac{g_{epi}}{G_0} \right) + \left(\frac{g_{epL}}{G_0} \right) \quad (85)$$

Donde la conductancia externa G_{exp} está considerada como el sumatorio de la parte ideal sin pérdidas de g_{epi} y una parte adicional con pérdidas g_{epL} . Por tanto, los coeficientes de acoplo totales se pueden considerar como una suma del acoplo sin pérdidas β_i y con pérdidas β_L .

$$\beta_1 = \beta_{1i} + \beta_{1L} \quad \beta_2 = \beta_{2i} + \beta_{2L} \quad (86)$$

Donde los coeficientes de acoplo con pérdidas y sin pérdidas se pueden calcular de la siguiente manera:

$$\beta_{pi} = \frac{g_{epi}}{G_0} = \frac{R_0 R_C}{(R_C + R_{sp})^2 + X_{sp}^2} \quad (87)$$

$$\beta_{pL} = \frac{g_{epL}}{G_0} = \frac{R_0 R_{sp}}{(R_C + R_{sp})^2 + X_{sp}^2} \quad (88)$$

Para resonadores en modo de reflexión, se ha demostrado que se pueden calcular las partes sin pérdidas y con pérdidas del coeficiente de acoplo a partir del diámetro en reflexión del círculo Q y del diámetro del círculo de pérdidas de acoplo. Se obtiene β_{pi} y β_{pL} , son términos para los resonadores en modo de transmisión que utilizan el mismo enfoque geométrico, pero resolviendo los cuatro componentes del coeficiente de acoplo en lugar de dos.

Se han determinado los diámetros de los círculos Q entendiendo que un punto desintonizado se encuentra diametralmente opuesto a un punto de resonancia para cualquier círculo Q, es decir:

$$\text{diámetro del círculo } Q = a_2 - \frac{a_1}{a_3} \quad (89)$$

Por lo tanto, el diámetro en transmisión S_{21} en el círculo Q, puede ser descrito como:

$$\text{diámetro del } S_{21} \text{ en el círculo } Q = \frac{2R_0 R_C |Y_{exp1} Y_{exp2}|}{(1+\beta_1+\beta_2)} \quad (90)$$

y el diámetro en reflexión del S_{11} o S_{22} del círculo Q se puede calcular como se muestra a continuación:

$$\text{diámetro } S_{pp} \text{ del círculo } Q = \frac{2R_C R_0 Y_{exp}^2}{(1+\beta_1+\beta_2)} = \frac{|2R_C R_0 Y_{exp}|^2}{(1+\beta_1+\beta_2)} \quad (91)$$

Si se sustituye la expresión (67) en (90) y (91) y conociendo la forma de la parte sin pérdidas del coeficiente de acoplo descrito en (87). Los diámetros de los círculos Q, en transmisión y reflexión, se pueden expresar en términos de los coeficientes de acoplo:

$$\text{diámetro } S_{21} \text{ del círculo } Q = \frac{2\sqrt{\beta_{1i}}\sqrt{\beta_{2i}}}{(1+\beta_1+\beta_2)} \quad (92)$$

$$\text{diámetro } S_{11} \text{ del círculo } Q = \frac{2\beta_{1i}}{(1+\beta_1+\beta_2)} \quad (93)$$

$$\text{diámetro } S_{22} \text{ del círculo } Q = \frac{2\beta_{2i}}{(1+\beta_1+\beta_2)} \quad (94)$$

Para un caso real, cuando las observaciones se hacen desde la entrada real del resonador, el acoplo asociado al círculo con pérdidas se puede visualizar como un círculo de resistencia constante que pasa por el punto de reflexión desintonizado $S_{11\alpha}$ del S_{11} del círculo Q y por el punto $\Gamma = 1 + j0$ en la carta de Smith. El diámetro del círculo de pérdidas de acoplo se puede encontrar como:

$$(\text{diámetro del círculo de pérdidas})_{\text{puerto } p} = \frac{1-S_{ppd}^2}{1-S_{ppd}\cos\phi} \quad (95)$$

Para obtener las ecuaciones que relacionan los diámetros de acoplo-pérdidas con las partes sin pérdidas y con pérdidas del coeficiente de acoplo, es necesario considerar que la ecuación para el diámetro del círculo de pérdidas para el puerto “p” es:

$$d_p = \frac{2}{1 + \frac{R_{sp}}{R_C}} \quad (96)$$

Si se relacionan las expresiones (88) y (90):

$$\frac{R_{SP}}{R_C} = \frac{g_{epL}}{g_{epi}} = \frac{\beta_{pL}}{\beta_{pi}} \quad (97)$$

Por lo tanto, el diámetro del círculo del acoplo con pérdidas se puede obtener como:

$$d_p = \frac{2}{1 + \frac{\beta_{pL}}{\beta_{pi}}} \quad (98)$$

Resolviendo (93), (94) y (97) para los coeficientes de acoplo β_{1i} , β_{1L} , β_{2i} y β_{2L} , produce las siguientes expresiones:

$$\beta_{1i} = \frac{x}{2 \left[1 - \left(\frac{x}{d_1} + \frac{y}{d_2} \right) \right]} \quad (98)$$

$$\beta_{1L} = \left(\frac{2}{d_1} - 1 \right) \beta_{1i} \quad (99)$$

$$\beta_{2i} = \frac{y}{2 \left[1 - \left(\frac{x}{d_1} + \frac{y}{d_2} \right) \right]} \quad (100)$$

$$\beta_{2L} = \left(\frac{2}{d_2} - 1 \right) \beta_{2i} \quad (101)$$

Donde d_1 es el diámetro del puerto 1 del círculo de acoplo-pérdidas, d_2 es el diámetro del puerto 2 del círculo de acoplo-pérdidas, “x” es el diámetro por puerto 1 (S_{11}) del círculo Q y “Y” es el diámetro por el puerto 2 (S_{22}) del círculo Q.

Usando las ecuaciones derivadas del artículo de K. Leong, los valores de los coeficientes de acoplo y el factor de calidad cargado (Q_L) de un resonador dieléctrico en modo de transmisión se pueden obtener a partir de un conjunto de datos de S_{11} , S_{21} y S_{22} medidos alrededor de la frecuencia de resonancia y aplicando procedimiento de ajuste por curva.

Capítulo 3: Desarrollo

Este tercer capítulo va a estar dividido en dos partes diferenciadas. Una primera parte donde se va a explicar el desarrollo del programa que permite la medida del factor de calidad de las cavidades resonantes y una segunda parte donde se va a hablar del desarrollo y fabricación de un resonador a 352 MHz.

3.1. Medida automatizada del factor de calidad en cavidades resonantes

El primer punto del TFM consiste en realizar un programa que mide el factor de calidad de una cavidad resonante. Para ello se ha empleado el lenguaje de programación Python. Para realizar el programa se ha utilizado el entorno de desarrollo integrado **Jupyter Notebook** que forma parte del paquete Anaconda. El código fuente del programa desarrollado aparece completo en el anexo del documento.

Debido a la extensión que presenta el programa, se ha realizado un diagrama de flujo del software que permite hacerse una idea resumida de cuáles son los puntos clave del programa. A continuación, se muestra el diagrama:

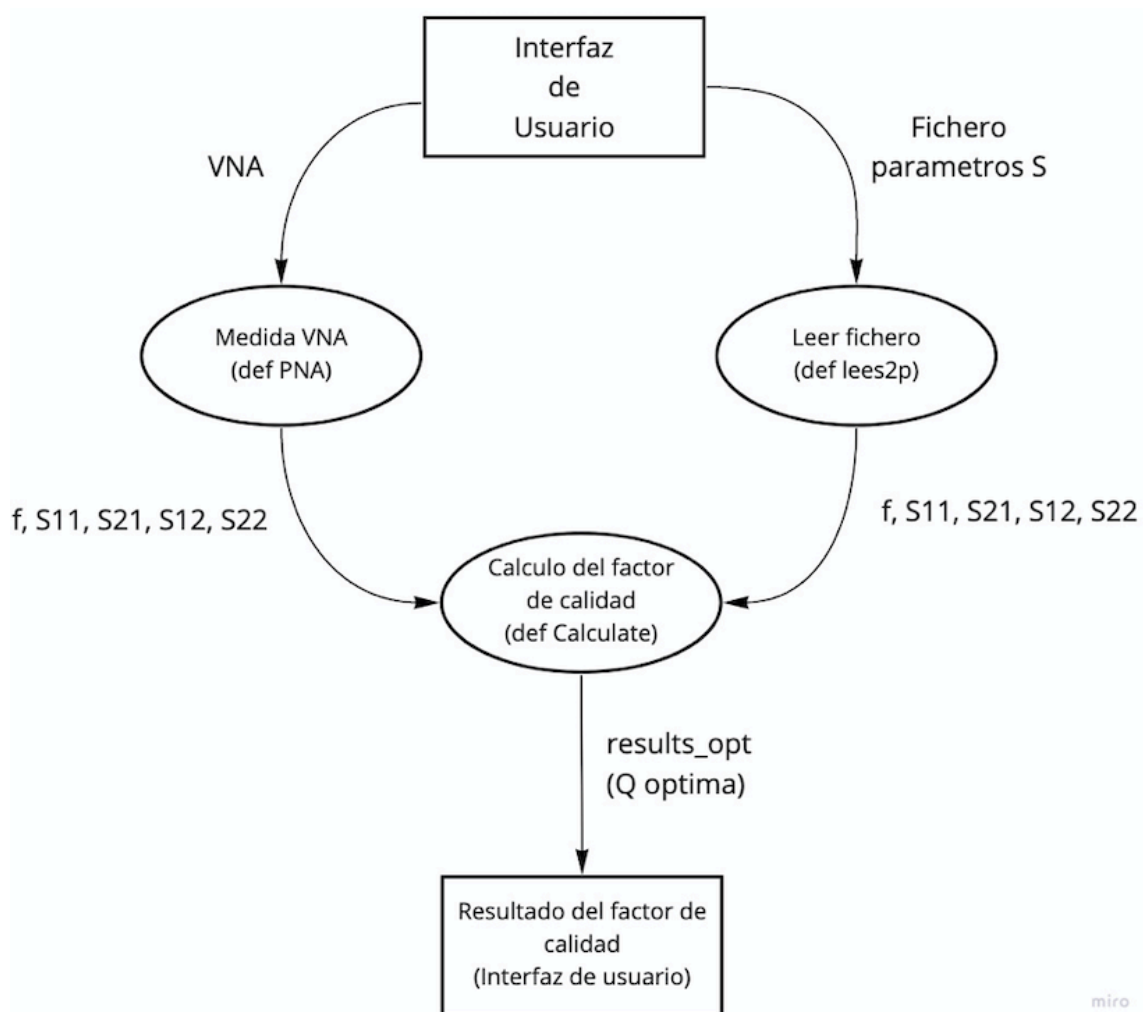


Figura 22. Diagrama de flujo del programa de medida de factor de calidad

Como se puede observar en la figura anterior, el programa cuenta con una interfaz de usuario para la introducción de datos y presentación de resultados, y donde se escoge, entre muchas otras cosas, si se va a utilizar el VNA para realizar medidas o bien un fichero de parámetros S procedente de una medida anterior o una simulación. Una vez obtenidos los valores los parámetros S con la frecuencia, es la función **“calculate”** la cual realiza el cálculo del factor de calidad de las tres maneras comentadas en la introducción del documento. Finalmente, se representa a través de la interfaz de usuario el resultado óptimo del factor de calidad.

A continuación, se va a realizar una descripción más detallada del programa y sus funciones.

Como se ha explicado anteriormente, el programa puede recibir los datos de entrada a partir de un fichero S2P o un analizador de redes. En el caso de escogerse la opción del fichero de parámetros S, la interfaz permite buscar y cargar el fichero en el árbol de directorios.

Otra de las funcionalidades que presenta es dar la opción al usuario de visualizar las gráficas obtenidas a lo largo del cálculo del factor de calidad. Para ello habría que pulsar en la opción **“Show Results”**. En el caso de que el usuario este interesado en ver los cálculos intermedios realizados a la hora de calcular el factor de calidad tendría que pulsar en la opción **“verbose”**. Una vez que el usuario ha escogido si quiere que el programa le proporcione los resultados intermedios, se pulsa el botón **“Calculate”** y el programa comienza con el cálculo del factor de calidad. Una vez que se ha finalizado el cálculo se muestra en la interfaz de usuario el factor de calidad calculado que es el cargado y el descargado, además de la frecuencia de resonancia de la cavidad y los coeficientes de acoplo.

Figura 23. Interfaz de usuario

Una vez que el usuario pulsa el botón **“calculate”** lo primero que hace el programa es ir a la función a través de la cual va a obtener los datos de entrada.

Si la opción que se ha escogido es la de obtener los datos de entrada a través del analizador de redes, el programa llamará a la función **“def PNA”**. Esta función se encarga de realizar la comunicación y recogida de datos del VNA, a través del protocolo VISA (Virtual Instrument Software Architecture), para lo cual se emplea la librería **pyvisa**. Una vez importada la librería, se realiza la conexión a través de la dirección GPIB indicada, que por defecto es la 16. Una vez realizada la conexión, se indica qué medida se quiere hacer en cada una de las trazas, en este programa interesa medir los diferentes parámetros S. A

continuación, se comprueba si el analizador de redes está calibrado o no (si aparece un 0 no está calibrado y si aparece un 1 el analizador está calibrado). Las siguientes líneas de código son las que permiten recoger los datos de las trazas definidas al principio del código. Los datos que se han recogido del analizador de redes están almacenados en un array (data) de tal manera que las siguientes líneas del código lo que hacen es sacar todos los valores de frecuencia y dividir los parámetros S medidos en su magnitud y fase, ya que los datos recogidos a través del analizador de redes son números complejos en formato módulo y fase. Para finalizar con esta función, se hace una conversión y se pone los parámetros S en formato real/imaginario y se representan en una gráfica para comprobar que queda de la misma manera que en el analizador de redes. Esta función tiene como parámetros de salida la frecuencia y el S_{11} , S_{22} , S_{21} y S_{12} .

Si en lugar de utilizar el VNA se escoge la opción de un fichero S2P, el programa llama a la función **“def call_file”**. Esta función se encarga de abrir el fichero con formato S2P escogido por el usuario, llamar a la función **“def lees2p”**, poner los parámetros S en todos los formatos con los que se va a trabajar a lo largo del programa (dB y fase) y representarlos. Los parámetros que devuelve esta función son: la frecuencia y el S_{11} , S_{22} , S_{21} y S_{12} . La función **“def lees2p”** es quien se va a encargar de poner los parámetros S en formato real/imaginario. Análogamente, esta función devuelve como parámetro de retorno la frecuencia y el S_{11} , S_{22} , S_{21} y S_{12} .

La siguiente función que se va a comentar es la **“def Calculate”**. Esta función es la encargada de realizar los cálculos de los factores de calidad tanto en reflexión como en transmisión para los puertos uno y dos. En primer lugar, se definen las variables que se van a utilizar a lo largo de la función y se crea un fichero **“resultados.txt”** donde se guardarán todos los factores de calidad calculados, sus frecuencias de resonancia y sus respectivos coeficientes de acoplo. Para realizar las medidas de los factores de calidad tanto en transmisión como en reflexión, se han definido unos umbrales que indican cuándo se trata de una medida en reflexión desde uno o desde dos puertos, y/o una medida en transmisión.

En el caso de reflexión, se ha encontrado que cuando dB(S_{11}) o dB(S_{22}) no presenta una variación mayor de 0.2 dB en el rango de frecuencias de medida, se puede considerar que no existe resonancia y por tanto no es conveniente realizar el cálculo del factor de calidad en reflexión ya que muy probablemente arrojará resultados erróneos.

Por su parte, se considera que dB(S_{21}) debe ser mayor de -70 dB para determinar que hay un pico de resonancia y entonces se debe realizar los cálculos de factor de calidad en transmisión.

Las siguientes líneas de código de la función **“calculate”** son las que se encargan de realizar el cálculo del factor de calidad. Primero se realiza la medida en reflexión para puerto 1.

La función **“Qu_refl”** es la encargada de calcular el factor de calidad en reflexión. En esta primera llamada calcula el factor de calidad por el método de los tres puntos y por el método de Kajfez para el puerto 1 ya que recibe de parámetro de entrada la frecuencia y el S_{11} . Comienza con el cálculo del factor de calidad utilizando el método de los tres puntos, para ello se llama a la función **“def searchresonance”** que es la encargada de buscar la resonancia y el valor desintonizado del S_{11} . Para realizar la búsqueda de la

resonancia se utiliza en primero una función de la librería **numpy** que localiza el valor y la posición del mínimo del S_{11} en dB. Una vez que se tiene el valor y la posición se utiliza la posición para obtener el valor del S_{11} y su frecuencia de resonancia. Siguiendo paso es localizar el valor desintonizado, para ello se utiliza esta vez la función “max” de numpy que de nuevo proporciona valor y posición donde el S_{11} es máximo en dB. De nuevo se utiliza la posición para obtener el valor del S_{11} desintonizado y su frecuencia. Estos valores calculados en la función son los que se devuelven como argumento de salida.

Una vez que se ha obtenido el S_{11} a la frecuencia de resonancia y desintonizado se llama a la función “def phaseoffset” que es la encargada de desplazar la fase del círculo Q a la recta real. En la figura 24 que se muestra a continuación se puede observar el desplazamiento que realiza esta función:

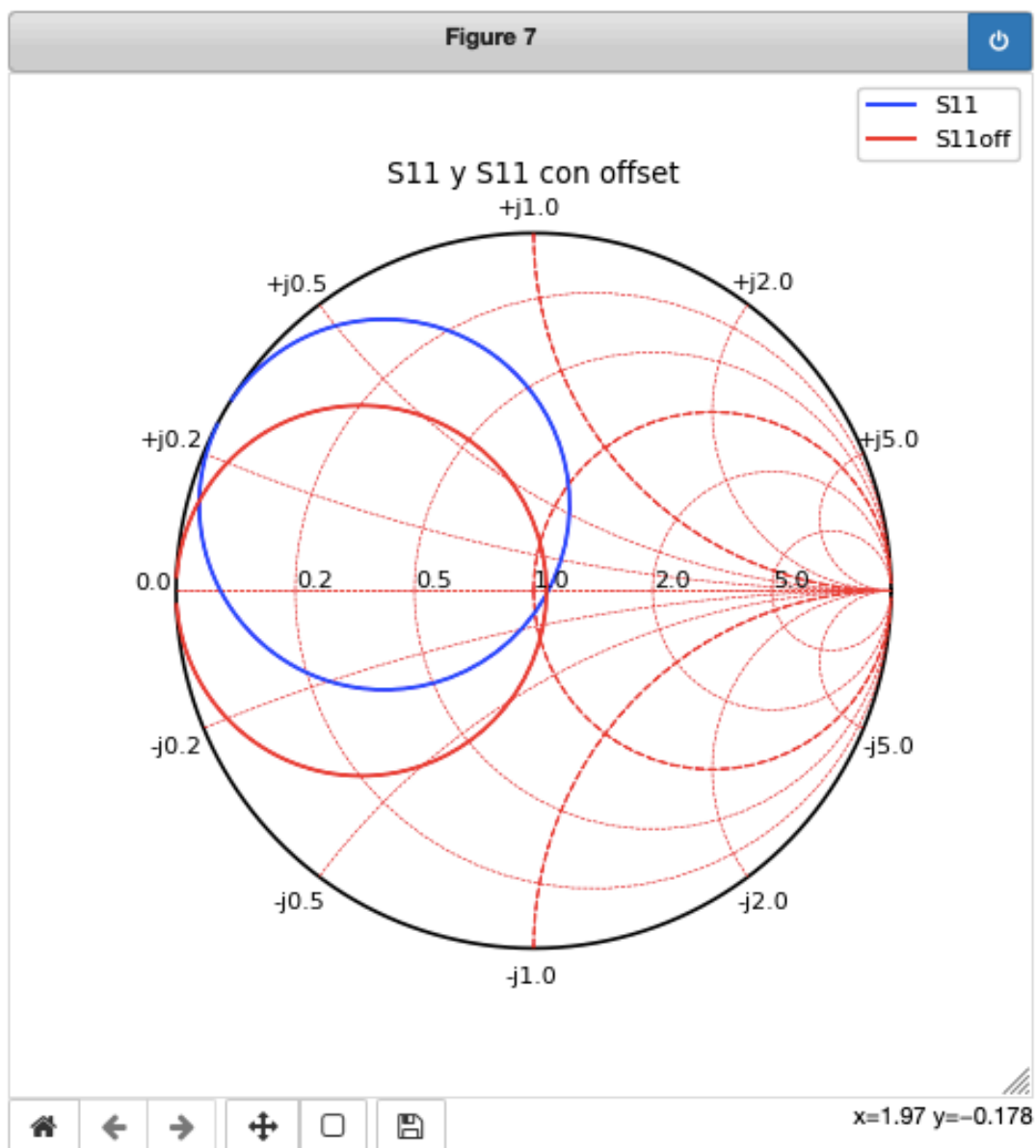


Figura 24. S_{11} a la frecuencia de resonancia y S_{11} desintonizado.

Una vez que se obtiene el S_{11} con la fase desplazada (S_{11off}) las siguientes líneas de código se encargan de buscar los puntos a $\pm 45^\circ$ en el S_{11} a las frecuencias más cercanas ($realS45$) y desplazar una posición a la izquierda los elementos del array S_{11off} (S_{11offd}). A continuación, el programa se queda con la parte real del S_{11off} y S_{11offd} . Después se buscan los índices que están justo por debajo de $realS45$ en el array original ($S_{11offreal}$) y por tanto en el array desplazado ($S_{11offdreal}$) queda justo por encima en la zona de pendiente positiva del S_{11off} , para ello se utiliza la función “**np.where**” de la librería **numpy** que devuelve una posición ($ind1$ y $ind2$). Una vez que se obtiene los índices comentados se realiza una interpolación para calcular de manera más exacta los valores de las frecuencias que se van a utilizar para calcular el factor de calidad cargado. En las siguientes líneas de código se muestra la interpolación que se ha llevado a cabo para calcular la f_1 y f_2 . Esta interpolación tiene en cuenta la posibilidad de que, por efecto de ruido en la medida, se produzcan varios pasos por $\pm 45^\circ$.

```
Af = f[1] - f[0]; # Ponemos un índice menor por Python
f1 = f[ind1[0]] + Af * (1 + ind1[-1] - ind1[0]) * (realS45 - S11offreal[ind1[0]]) / (S11real[ind1[-1]+1] - S11real[ind1[0]]);
f2 = f[ind2[0]] + Af * (1 + ind2[-1] - ind2[0]) * (realS45 - S11offreal[ind2[0]]) / (S11real[ind2[-1]+1] - S11real[ind2[0]]);
```

Figura 25. Interpolación realizada en Python

Para finalizar con el método de los tres puntos, se van a utilizar las fórmulas (22) y (23), vistas en la introducción teórica, para calcular el factor de calidad cargado y descargado y la función “**def checkundercoupling**” para comprobar si el resonador está sobre acoplado o infracoplado. La “**def checkundercoupling**” compara el valor de $S_{11offreal}$, en la posición de resonancia, con un cero. Si el valor es menor o igual que cero la función devuelve como parámetro un 1 lo cual significa que el coeficiente de acoplo está infracoplado y si no devuelve un 0, que implica que está sobreacoplado. De vuelta en la función “**def Qu_refl**” en función del resultado obtenido en “**def checkundercoupling**” se calcula el coeficiente de acoplo de una manera u otra.

El siguiente método que se utiliza en la función “**def Qu_refl**” es el de Kajfez. Como se comentó en la parte teórica, hay que hacer el ajuste dos veces, por tanto, se comienza el método con un ciclo “**for**” que resolverá dos veces el sistema lineal. Dentro del ciclo, se restringe el ancho de banda ($spank$), se busca el índice ($indk$) para el cual la frecuencia es mayor que la frecuencia menos el ancho de banda reducido y a su vez la frecuencia es menor que la frecuencia más el ancho de banda reducido, para realizar dicha búsqueda se ha utilizado la función “**np.where**” de la librería **numpy**. Una vez localizada la posición, el código continúa cogiendo la frecuencia (fk) y el S_{11} ($s11k$) correspondiente a dicho índice.

Siguiente paso es generar la matriz A y la matriz b y resolver el sistema.

```
x = np.linalg.lstsq(A, b, rcond=None)
s = x[0]
a1 = s[0]; a2 = s[1]; a3 = s[2]
```

Figura 26. Resolución del sistema en “**Qu_refl**”

Para resolver el sistema se ha utilizado el método `linalg.lstsq` de la librería **numpy** y es una función que obtiene la solución de mínimos cuadrados para un sistema de ecuaciones lineales (expresado en forma matricial). Una vez resuelto el sistema la solución para a_1 ,

a_2 y a_3 se obtienen a partir del primer array de soluciones de “X”. A continuación, se obtiene el factor de calidad descargado, el diámetro del círculo, Γ_s , Γ_i , factor de calidad cargado y el coeficiente de acoplo, para ello se han utilizado las expresiones (47), (50), (51), (52) y (53) vistas en la parte teórica.

Una vez que se tiene calculado el Γ_s se comprueba si su valor es menor que 0.99, si es así, se calcula de nuevo el diámetro del círculo Q y el valor de ω_L . Para calcular ω_L se realiza la diferencia entre Γ_i y Γ_s (*difRho*), después el programa se queda con el valor positivo y lo pasa a dB. Una vez que se tiene en dB el “difRho” se utiliza la función “max” de numpy para calcular de nuevo la posición y el valor de ω_L . Una vez que se ha obtenido la posición, se coge el valor de Γ_i para esa frecuencia (Rho1) y se calcula utilizando Pitágoras el ángulo Φ . Una vez que se ha obtenido Φ , se calcula α como la suma interna de triángulos y finalmente se calcula el diámetro d_2 del círculo que toca la circunferencia de Smith. Con d_2 se calcula el nuevo coeficiente de acoplo, el factor de calidad descargado y el ancho de banda. Como parámetro de retorno de la función “Qu_refl” se devuelve los resultados obtenidos por el método de los tres puntos, los de Kajfez, el diámetro del círculo Q (d), el S_{11} , el S_{11} con la fase desplazada (s11off) y el d_2 .

Una vez medido el factor de calidad en reflexión para el puerto 1, se escribe los obtenidos en el fichero resultados y por pantalla. Se guarda el resultado correspondiente al método de kajfez como resultado óptimo.

La siguiente medida que realiza el programa, en la función “def calculate”, es en reflexión para el puerto 2. De nuevo para realizar la medida del factor de calidad el máximo del S_{22} menos el mínimo del S_{22} debe tener un valor mayor al umbral definido (dBS22th). De la misma manera que se hizo en la medida anterior, una vez que se cumple que la diferencia es mayor que el umbral se llama a la función “Qu_refl” y se realizan los mismos cálculos, pero esta vez en vez de usar el S_{11} , se utiliza el S_{22} . El resultado se escribirá en el fichero resultados, en el caso de ser el resultado óptimo se escribirá en la variable “results_opt” y también por pantalla por pantalla.

Cuando se realizan los cálculos de factor de calidad con medidas en reflexión para los dos puertos, ocurre que también hay transmisión entre ambos puertos y acoplo cruzado, por lo que es necesario aplicar la corrección presentada en las ecuaciones (26) y (27). Si se cumple con la condición, el programa llama a la función “def Qu_r_t_correction” pasándole como parámetros de entrada todos los resultados obtenidos en las medidas en reflexión para el puerto 1 y el puerto 2. En la función “def Qu_r_t_correction” se recalcula para el puerto 1 y 2: el coeficiente de acoplo con la expresión (102), las Q descargadas y los anchos de banda. Esta corrección es necesaria en las cavidades que presentan dos acoplos. Cuando una cavidad presenta más de un acoplo no despreciable, se afectan unos a otros pareciendo más pequeños de lo que realmente son, por lo que es necesario realizar la corrección. Los parámetros que devuelve la función son las frecuencias de resonancia, los anchos de banda, las Q cargadas, los coeficientes de acoplo y las Q descargadas para cada uno de los puertos.

$$K_1 = \frac{k_1(1+k_2)}{(1-k_1k_2)} \quad K_2 = \frac{k_2(1+k_1)}{(1-k_1k_2)} \quad (102)$$

Una vez calculado el factor de calidad en reflexión aplicando una corrección, se escriben los resultados obtenidos en el fichero resultado y por pantalla.

La última medida que se realiza en la función “def calculate” es en transmisión. Al igual que ocurre en las medidas anteriores, para realizarla hay que cumplir con la condición de que el máximo del S_{21} sea mayor que el umbral definido al principio de la función (dBS21th).

Una vez dentro de la condición, el programa llama a la función que se encarga de calcular el factor de calidad en transmisión, la función es “def Qu_trans”. Los parámetros de entrada que hay que proporcionarle a la función son: la frecuencia, el S_{11} , S_{22} y el S_{21} . El primer método que se aplica es el de los tres puntos, que al igual que en la función “def Qu_refl” se utiliza la función “max” de la librería numpy para localizar la posición y el valor de la frecuencia de resonancia para el S_{21} . Una vez localizada se guarda la frecuencia de resonancia en la variable “f0”. A continuación, se busca el ancho de banda 3dB (G3dB) y se genera un nuevo array con los mismos valores que tiene S_{21} en dB con la diferencia que este desplazado una posición hacia la derecha los valores del array (dBS21d). Las siguientes líneas de código se encargan de buscar los índices que cumplan la condición que se muestra en la figura 27 (ind1 y ind2). Para realizar esta búsqueda se ha usado la función “np.where” de la librería numpy.

```
#find permite buscar indices y valores de elementos no nulos y te da o localiza su posicion
ind1 = np.where((dBS21 < G3dB) & (dBS21d > G3dB)) #equivalente al find de matlab
#ind1 = ind1 ((ind1 < indres).nonzero())
ind2 = np.where((dBS21 > G3dB) & (dBS21d < G3dB))
#ind2 = ind2 ((ind2 > indres).nonzero()) #PUEDE ESTAR MAL
```

Figura 27. Búsqueda de los índices

Una vez calculados los índices, se realiza una interpolación para calcular los valores de f_1 y f_2 . En la figura 28 se puede observar la interpolación que se ha realizado para calcular los valores comentados.

```
Af = f[1] - f[0]
f1 = f[ind1[0]] + Af * (1 + ind1[-1] - ind1[0]) * (G3dB - dBS21[ind1[0]]) / (dBS21[ind1[-1]+1] - dBS21[ind1[0]])
f2 = f[ind2[0]] + Af * (1 + ind2[-1] - ind2[0]) * (G3dB - dBS21[ind2[0]]) / (dBS21[ind2[-1]+1] - dBS21[ind2[0]])
..
```

Figura 28. Interpolación.

A partir de obtener f_1 y f_2 , se calcula el ancho de banda, la Q cargada (Q_L) y una aproximación del valor de Q descargado. El siguiente paso que realiza el código en la función “def Qu_trans” es calcular los acoplos tanto de entrada como de salida (VSWR1 y VSWR2) y finalmente calcular el valor de Q descargado en función de si las betas estas infra acopladas o sobre acopladas. Al igual que en la función “def Qu_refl” para saber si el coeficiente de acoplo está infra o sobre acoplado se llama a la función “def checkundercoupling” que es la encargada de realizar este cálculo y devolver un 1 o un 0 en función de cuál sea el resultado, como se ha explicado anteriormente. A continuación, se calcula la β_1 y la β_2 como se muestra en la figura 29.

```

if undercoupled1 == 1:
    beta1 = 1 / VSWR1
else:
    beta1 = VSWR1

undercoupled2 = checkundercoupling(f, S22, verbose)

if verbose:
    print("undercoupled_trans2: ", undercoupled2)

if undercoupled2 == 1:
    beta2 = 1 / VSWR2
else:
    beta2 = VSWR2

```

Figura 29. Cálculo de las betas en función de si es sobre/infra acoplado.

Para finalizar con este primer método se aplica la expresión (101) para calcular los coeficientes de acoplo aplicándoles el factor corrector y se calcula el valor de Q descargada (Q_u) final utilizando la expresión (28).

Una vez que se ha realizado el método de los tres puntos, el siguiente que se calcula en esta función es el método de K. Leong. Al igual que en la función “def Qu_refl” se resuelve dos veces el sistema correspondiente al curve fitting. La primera parte de este método es exactamente igual a la explicada en “def Qu_refl”, se comienza restringiendo el ancho de banda (spank), se busca el índice (indk) para el cual la frecuencia es mayor que la frecuencia menos el ancho de banda reducido y a su vez la frecuencia es menor que la frecuencia más el ancho de banda reducido, para realizar dicha búsqueda se ha utilizado la función “np.where” de la librería **numpy**. En la figura 28 se muestra la condición que presenta la función “np.where”. Una vez localizado el índice (indk) se recoge el valor de la frecuencia para ese índice en la variable “fk” y lo mismo se hace con el S_{21} para ese índice, el valor se recoge en la variable “S21k”.

```

spank = 3*BW
indk = np.where((f > (fl-spank/2)) & (f < (fl+spank/2)))
fk = f[indk]
S21k = S21[indk]
t = 2 * (fk - fl)/f0

```

Figura 30. Función np.where

A continuación, se generan las matrices A y la matriz b y se resuelve el sistema de nuevo, usando la función que convierte la resolución por mínimos cuadrados en una matriz lineal, que tiene la librería numpy y que se puede observar en la figura 26. Una vez obtenidos los valores de a_1 , a_2 y a_3 se puede calcular Γ_s con la expresión (57), el factor de calidad cargado con la expresión (83) y el diámetro del círculo Q con la expresión (88).

Después, se genera un vector “t” de 1601 valores cuyo principio y final son los mismos que los del vector generado en la figura 30. Una vez que se obtiene t, se calcula el Γ_i aplicando la fórmula (46). A continuación, se pasa a dB el valor de Γ_i y se realiza la búsqueda de su frecuencia de resonancia para ello se usa la función “max” de la librería

numpy que permite encontrar la posición Γ_i máximo y a partir de ahí se localiza la frecuencia de resonancia (f_0). Una vez localizada la frecuencia de resonancia se calcula el ancho de banda de este segundo método utilizando la fórmula que se muestra a continuación:

$$BW = \frac{f_0}{Q_L} \quad (102)$$

Una vez que se finaliza con el “for” anterior, al igual que ocurre en la función “def calculate”, si se cumple con el umbral dBS11 y el dBS22, entonces se llamara a la función “def Qu_refl” donde se llevara acabo de nuevo el método de los tres puntos y kajfez. Esta llamada a la función se realiza para obtener el valor del diámetro del círculo Q (dS11 y dS22) y el valor del diámetro d_2 (dl1 y dl2) correspondiente a cada puerto. De esta manera se calcularán a continuación las betas y el valor de Q descargada usando las expresiones (63) y (103).

$$\beta_1 = \frac{dS11*dl2}{dl2*dl1-dS22*dl1-dS11*dl2} \quad \beta_2 = \frac{dS22*dl1}{dl2*dl1-dS22*dl1-dS11*dl2} \quad (103)$$

En el caso de no cumplir con el umbral para alguno de los dos puertos se da un valor de cero al diámetro del círculo Q y un valor de 2 al d_2 . De la misma manera que cuando se ha llamado a “def Qu_refl” se calcula las betas y el factor de calidad descargado. En la figura a continuación, se muestra la llamada a la función “def Qu_refl” si cumple con el umbral y en el caso de no cumplir con el umbral los valores que toman el diámetro del círculo Q y el diámetro d_2 .

```
if (max(dBS11) - min(dBS11)) > dBS11th:
    rqrefl11,rqrefl11K, dS11, S11, S11off, dl1 = Qu_refl(f, S11,"in", verbose)
else:
    dS11 = 0
    dl1 = 2
if (max(dBS22) - min(dBS22)) > dBS22th:
    rqrefl22,rqrefl22K, dS22, S22, S22off, dl2 = Qu_refl(f, S22,"in", verbose)
else:
    dS22 = 0
    dl2 = 2
```

Figura 31. Llamada a "def Qu_refl" para calcular los diámetros necesarios.

De vuelta en la función “def calculate” se guardan los valores calculados en el fichero resultados y se representan por pantalla. Además, la variable “results_opt” se actualiza cogiendo el valor correspondiente al método de K. Leong, esto ocurre debido a que este método proporciona la mejor aproximación ya que hace curve fitting, además para calcular el valor de Q descargado final necesita realizar también Kajfez, por tanto, es una aproximación muy acertada del valor del factor de calidad.

Una vez que la función “def calculate” se ha ejecutado entera, se devuelve la variable “results_opt” a la interfaz de usuario. Finalmente, aparece el resultado óptimo por pantalla para que el usuario pueda verlo desde la interfaz de usuario, además de haberse generado un fichero donde aparecen todos los cálculos realizados a lo largo del programa. En la figura 32 se puede observar un ejemplo de como se muestra por pantalla el resultado óptimo final.

ESS BILBAO: QUALITY FACTOR MEASUREMENT

Input Data: ☐ VNA ☒ Input File

File: Parametros S Resonador.s2p

☐ Show Results

☐ History

☐ Verbose

Output Data:

f0 (MHz): 361,703125

Qo: 187

Qi: 100

beta1: 0,8613

beta2: 0

Figura 32. Resultado del factor de calidad medido para un resonador.

Solo ha quedado una función por comentar, que es la encargada de representar. Esta función se llama “**def smithchart**” y es la encargada de representar en una carta de Smith lo que en el programa sea necesario.

3.2. Fabricación de un prototipo de resonador

El siguiente punto por realizar en el TFM es el diseño y fabricación de un prototipo de resonador $\frac{\lambda}{4}$ a una frecuencia cercana a 352 MHz para la realización de pruebas de caracterización con el analizador de redes. Para el diseño del resonador se ha utilizado la herramienta ADS (Advanced Design System) de Keysight.

Los componentes que se van a utilizar inicialmente para realizar el resonador serán: un condensador, un GAP y un resonador de cuarto de onda realizado mediante un cable coaxial que irá soldado a tierra en su extremo. En primer lugar, se realiza un esquemático donde se han utilizado componentes ideales. Para conocer el valor de los componentes que se van a utilizar en este esquemático se han realizado ciertos cálculos. Primero se ha calculado la longitud que inicialmente va a presentar el cable. A continuación, se muestran los cálculos realizados:

$$L = \frac{\lambda}{4} \quad \lambda_0 = \frac{c}{f} = \frac{3 \times 10^8}{352 \times 10^6} = 80 \text{ cm} \quad \lambda_c = \frac{V_p}{f} = \frac{c}{\sqrt{\epsilon_r} f} = 60 \text{ cm} \quad \rightarrow \frac{\lambda}{4} = 15 \text{ cm} = L$$

$$\epsilon_{r-\text{teflon}} = 2 \quad V_p = \frac{c}{\sqrt{\epsilon_r}} \quad \rightarrow \frac{1}{\sqrt{\epsilon_r}} = 0.7 \text{ (70\%)}$$

Además de indicar la longitud del cable, hay que indicar ciertos valores correspondientes a la línea coaxial para intentar que la simulación sea lo más real posible. En la figura 31 se puede observar todos los valores utilizados para definir el coaxial de la manera más aproximada al que se va a utilizar en la fabricación.

Para conocer el valor que presentara el condensador, se ha realizado un esquemático donde se realiza un barrido de frecuencia desde los 100 MHz hasta los 500 MHz. En dicho esquemático aparecen el condensador y el cable a tierra. Una vez realizado el barrido de frecuencia se observa la frecuencia de resonancia que presenta el circuito y se utiliza la herramienta “Tune”, esto permite variar el valor de la capacidad hasta que la frecuencia de resonancia sea de 352 MHz. El valor del GAP se calcula de la misma manera que se ha calculado el valor del condensador. Finalmente, el circuito con todos los componentes ideales queda como se muestra en la figura siguiente:

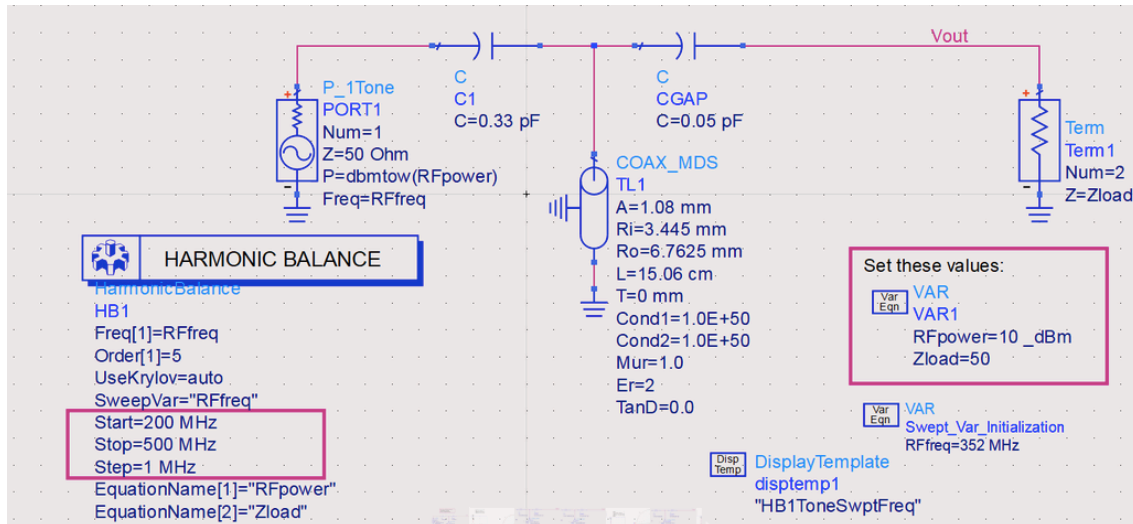


Figura 33. Circuito con componentes ideales.

Como se puede observar en la figura 33, se realiza un barrido de frecuencia desde los 200 MHz hasta los 500 MHz, para comprobar que el resonador ideal está trabajando a la frecuencia deseada, es decir a 352 MHz. En la figura a continuación, se muestra el resultado de la simulación:

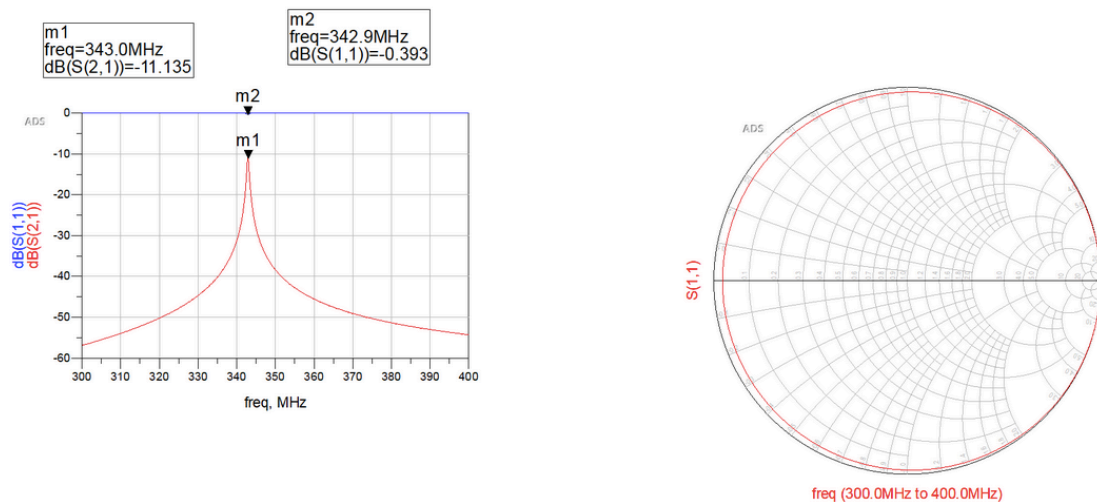


Figura 34. Resultado de la simulación del circuito ideal.

Como se puede observar en la figura 34, donde se exponen los resultados de este primer diseño, la frecuencia de resonancia es de 343 MHz, esto ocurre porque el S_{11} que presenta el circuito no es correcto. Si se observa la carta de Smith que aparece a la derecha de la figura, se puede ver que el S_{11} presenta sobre acoplo. Por tanto, se realiza una nueva simulación, esta vez utilizando la herramienta “tune” tanto en el valor del condensador como en la longitud del coaxial, para intentar que el S_{11} presente un acoplo crítico o por lo menos lo más cercano al centro. En la figura a continuación se muestra el nuevo circuito para estos cambios realizados.

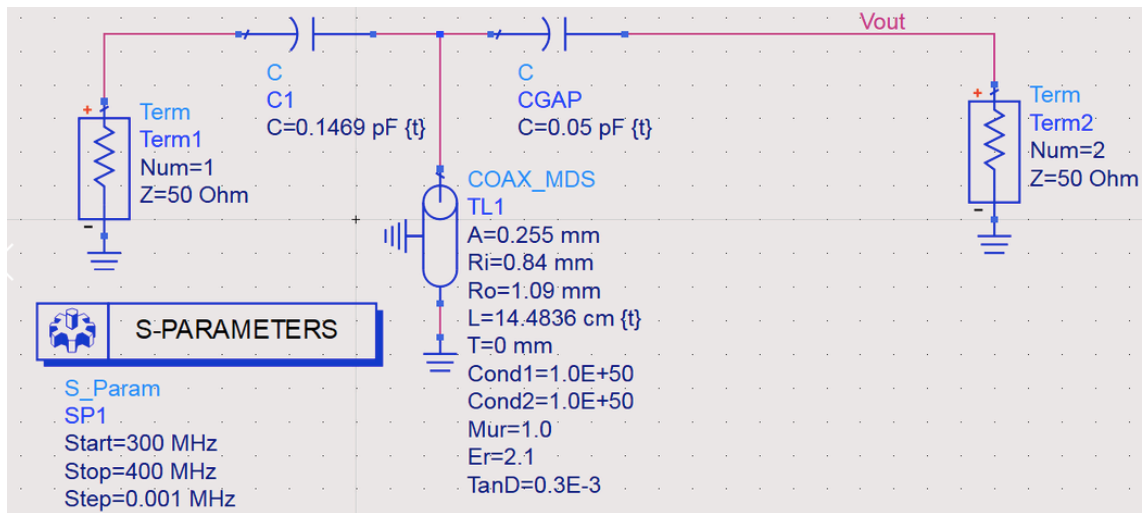


Figura 35. Circuito resultante de utilizar la herramienta "tune"

Como se puede observar, el valor del primer condensador ha disminuido de 0.33 pF hasta los 0.15 pF. La longitud del cable también ha disminuido, ha pasado de tener un valor de 15 cm a presentar una longitud de 14.5 cm. El resultado de simular este último circuito se muestra a continuación:

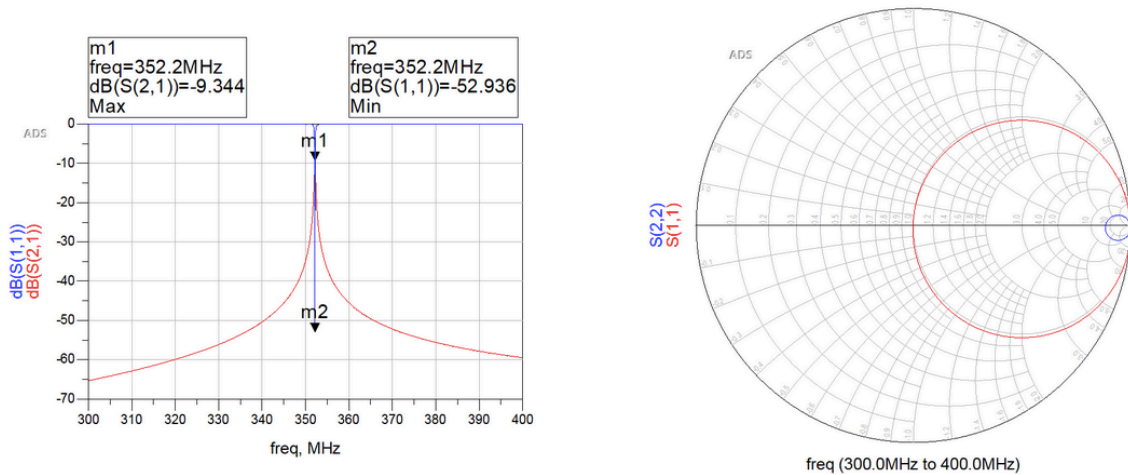


Figura 36. Resultado de la simulación del circuito tras utilizar la herramienta "tune"

Observando la figura 36, se puede ver como una vez realizado un ajuste de parámetros del circuito anterior, la frecuencia de resonancia es la buscada, es decir, 352 MHz. Además, al contrario que en la primera simulación, el circuito presenta adaptación a la entrada. El siguiente punto para observar es la carta de Smith que aparece a la derecha de la figura, en ella se puede observar que el S_{11} presenta un acoplo crítico, que es lo que se estaba buscando y que en la primera simulación no se había conseguido.

Una vez que el circuito ideal funciona correctamente, se ha realizado el diseño del circuito con componentes reales, para ello se ha utilizado el sustrato **Rogers 4003** [17] de 60 mils de espesor (1.524 mm). En las figuras a continuación, se muestra la definición del sustrato en ADS y el esquemático con componentes reales.

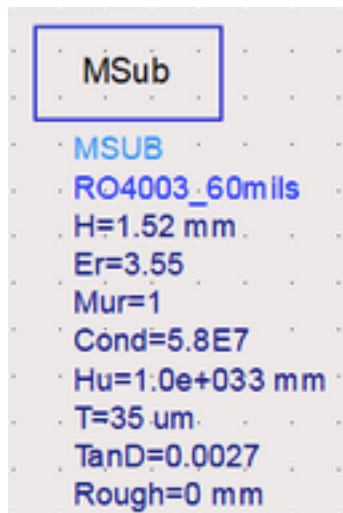


Figura 37. Definición del sustrato.

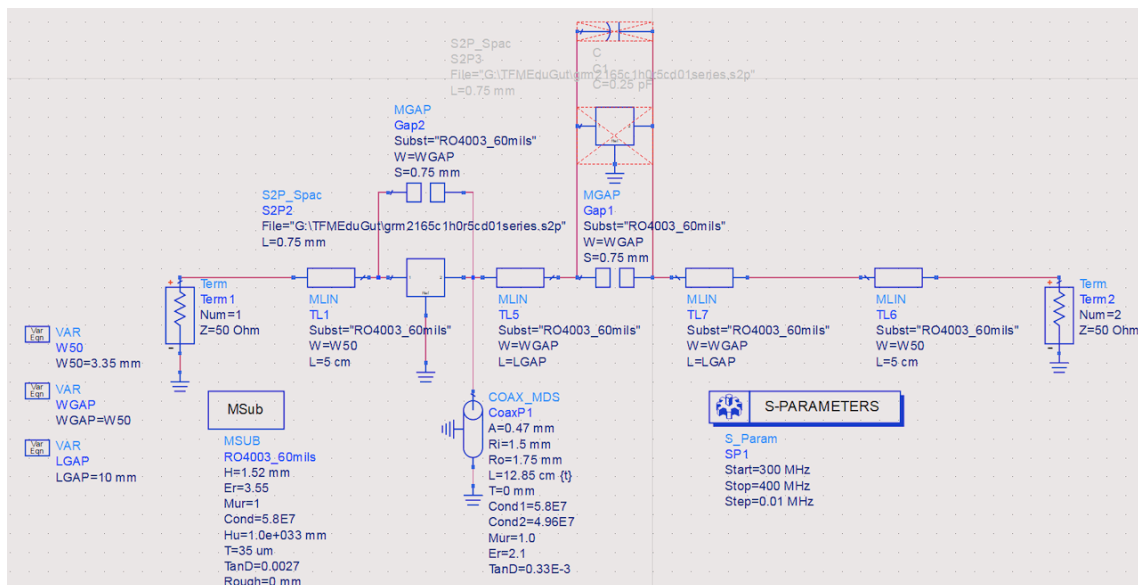


Figura 38. Esquemático correspondiente al resonador real.

Como se puede observar en la figura 38, el esquemático con componentes reales muestra algún cambio respecto al esquemático con elementos ideales. Si se analizan los cambios, de izquierda a derecha, en primer lugar, se ha cambiado el condensador de 0.14 pF por un condensador de 0.5 pF aproximadamente en paralelo con un GAP, que presenta una separación de 0.75 mm. Los parámetros S del condensador se buscaron en el fabricante **Murata**, intentando utilizar alguno de un valor cercano al que se podía encontrar en el laboratorio. El siguiente cambio significativo que se ha llevado a cabo, es el cambio del GAP de 0.05 por uno con una separación de 0.75 mm. Estos cambios se han realizado tras añadir las líneas microstrip, realizar una simulación y observar que tanto la frecuencia de resonancia como la adaptación se habían visto modificadas notablemente.

A continuación, se muestra los resultados obtenidos al realizar las modificaciones:

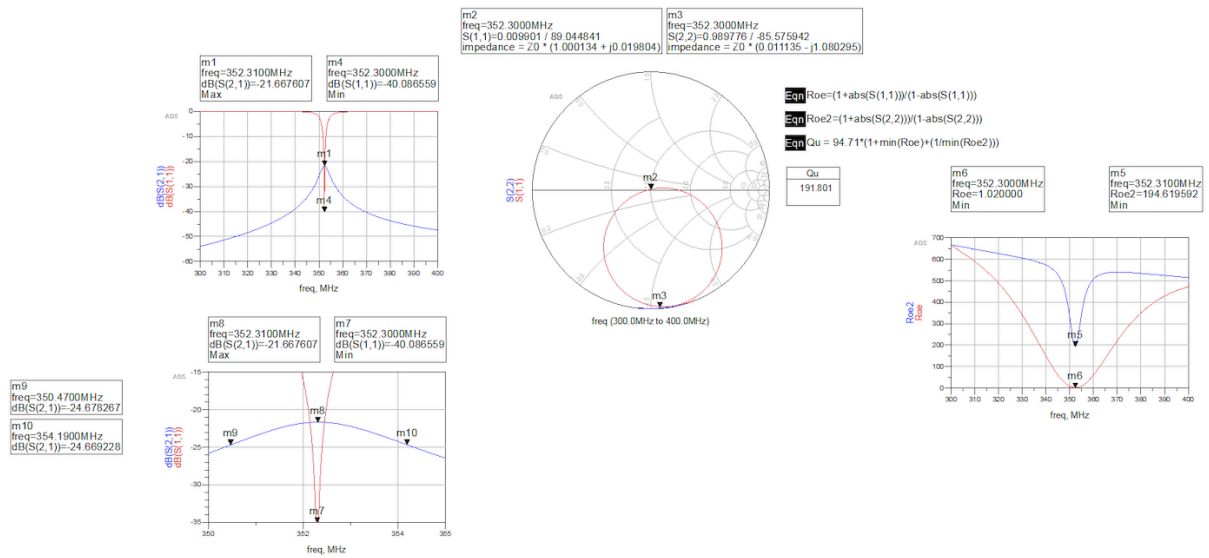


Figura 39. Simulación componentes reales.

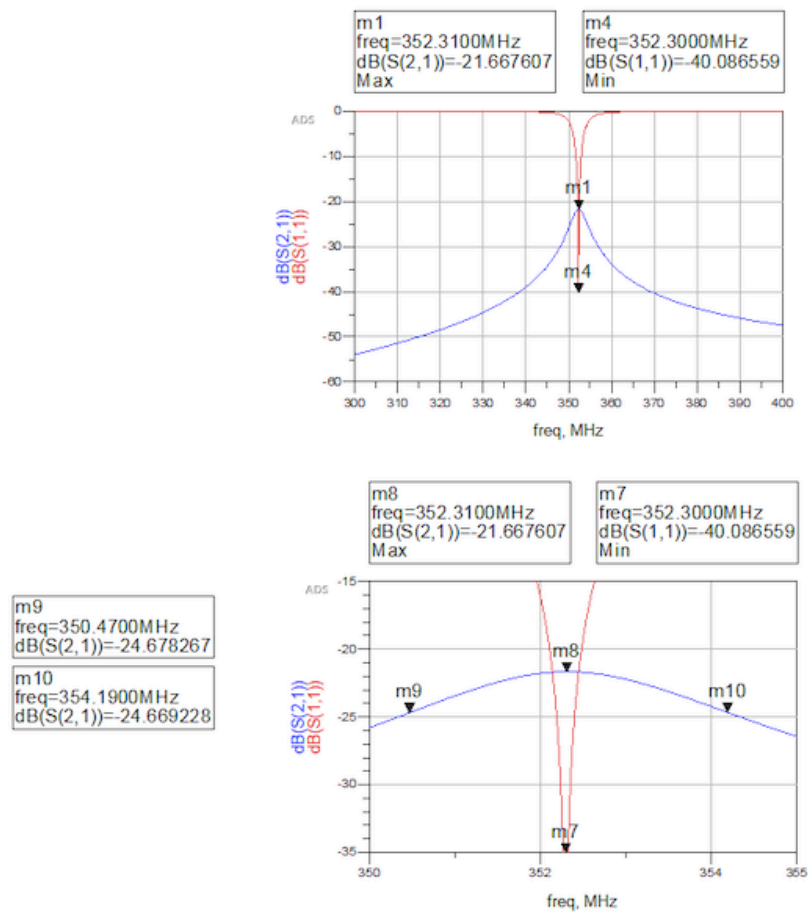


Figura 40. Simulación con componentes reales

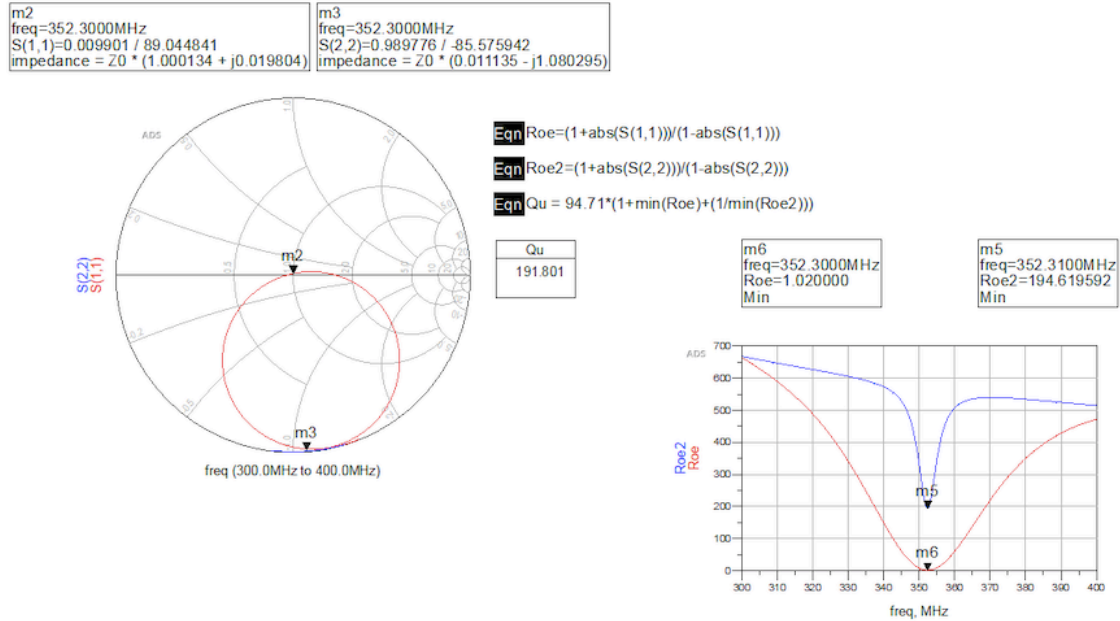


Figura 41. Simulación con componentes reales

Analizando los resultados de la figura 39 o más cerca en las figuras 40 y 41, se puede observar que la frecuencia de resonancia obtenida para este circuito es la deseada, es decir, 352 MHz y que a dicha frecuencia presenta una buena adaptación de entrada. Si analizamos la carta de Smith que aparece en la figura, se puede observar que el resonador presenta un acoplo crítico. Además, se ha utilizado el método de los tres puntos para realizar un cálculo aproximado del factor de calidad descargado que tiene que presentar el resonador, el valor de Q_u que presenta es de 191. Finalmente se ha representado los valores de ROE, que previamente se habían calculado para poder realizar el cálculo del factor de calidad a partir de la siguiente expresión:

$$ROE_1 = \frac{1 + |S_{11}|}{1 - |S_{11}|} \quad ROE_2 = \frac{1 + |S_{22}|}{1 - |S_{22}|} \quad Q_u = \frac{Q_L(1 + \min(ROE_1))}{\frac{1}{\min(ROE_2)}} \quad \text{donde } Q_L = 94.7$$

Una vez que el resonador trabaja a la frecuencia de resonancia deseada y presenta acoplo crítico como se ha podido confirmar en la figura 37, se ha realizado el trazado (layout) del circuito como paso previo a su fabricación. Para realizar este paso se han utilizado diferentes herramientas. En primer lugar, se ha usado la opción de generar layout automáticamente que presenta ADS, una vez que el layout queda generado se ha exportado como ".dxf" y se ha utilizado la herramienta **LibreCAD** para colocar las vías que permitirán al coaxial del resonador estar conectado a tierra por uno de sus extremos. En la imagen a continuación, se muestra el layout final:

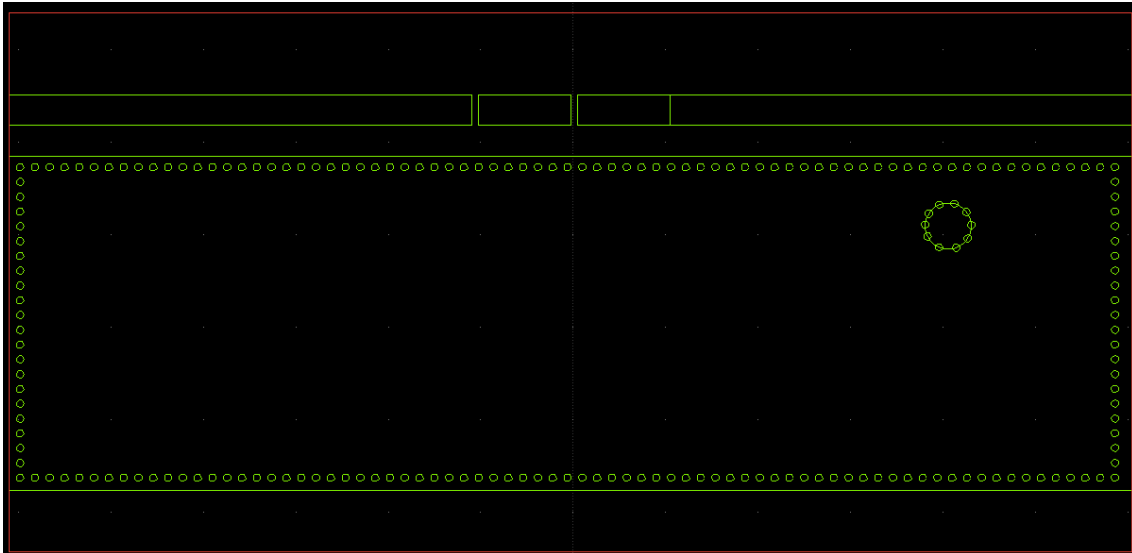


Figura 42. Layout final

Finalizado el layout, se fabricó el resonador. Una vez montada la placa, se conectó el coaxial, para ello se soldó el conductor central de uno de los extremos del resonador coaxial a la segunda línea microstrip, vista la placa desde la izquierda. El otro extremo del coaxial se conectó de la siguiente manera: se soldó el conductor central a la malla del conductor externo para formar un cortocircuito entre ambos. Este extremo se soldó a su vez a tierra a través la zona de vías metalizadas. En la siguiente figura se muestra el resultado final que presenta el resonador fabricado:



Figura 43. Resonador fabricado.

En la siguiente figura se muestra la conexión en paralelo del primer condensador y con el primer GAP. Aunque en el layout esta conexión no se había previsto, una vez que se ha realizado la fabricación del resonador, dada la pequeña dimensión del GAP, es posible soldar un condensador de 0.5 pF. Este último paso, puede generar una variación en la frecuencia de resonancia ya que el condensador con el que se contaba no era exactamente del mismo valor que el utilizado durante la simulación. En el capítulo de resultados comentara si la frecuencia de resonancia ha sido la deseada o se ha visto desplazada tras la fabricación.



Figura 44. Condensador en paralelo con el primer GAP.

Capítulo 4: Resultados

En este cuarto capítulo se va a describir los resultados obtenidos al medir el factor de calidad del resonador diseñado y fabricado y de una cavidad resonante Buncher.

En primer lugar, se realizó la medida del factor de calidad del resonador $\frac{\lambda}{4}$, para ello se acudió al laboratorio y se conectó tanto el resonador como el ordenador al analizador de redes (VNA, Vector Network Analyzer). Para conectar el ordenador se utilizó un conversor GPIB-USB de HP (Hewlett-Packard). Una vez que se ha realizado las conexiones, se ponen las 4 trazas, cada una con un parámetro S en formato LOG/MAG y se busca el primer pico de resonancia y después se calibra el VNA. Tras realizar la calibración en el analizador de redes se ve lo mismo que en la siguiente figura:

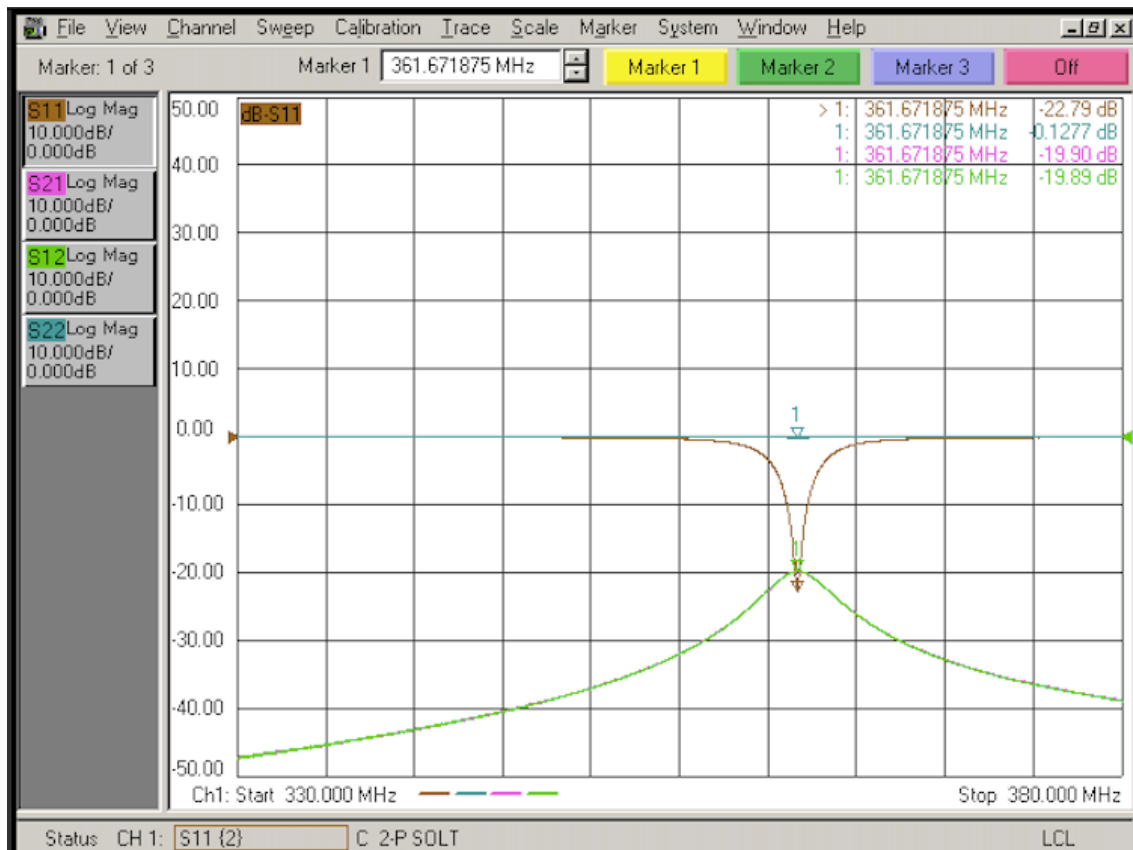


Figura 45. Resonador a 361 MHz.

Como se puede observar en la figura anterior, el resonador presenta la frecuencia de resonancia desplazada a 362 MHz en lugar de en 352 MHz, como se ha comentado en el capítulo anterior, este desplazamiento era de esperar. Para corregir esta diferencia de 10 MHz, se intentó añadir al circuito un condensador de pequeño valor en paralelo con el que forma parte del circuito original. Finalmente, como no se encontró ningún condensador lo suficientemente pequeño, se decidió dejar la frecuencia y realizar las medidas a esa frecuencia de resonancia.

Se puso en marcha el programa de Python para realizar la medida del factor de calidad, pero previamente se decidió utilizar el VNA para realizar la medida del factor de calidad manualmente y después comprobar que el resultado que proporcionaba el programa era correcto, debido a que era la primera medida que se realizaba con el programa. En la

figura a continuación, se muestra el resultado del factor de calidad medido por el propio analizador de redes con su utilidad de marcador Q.

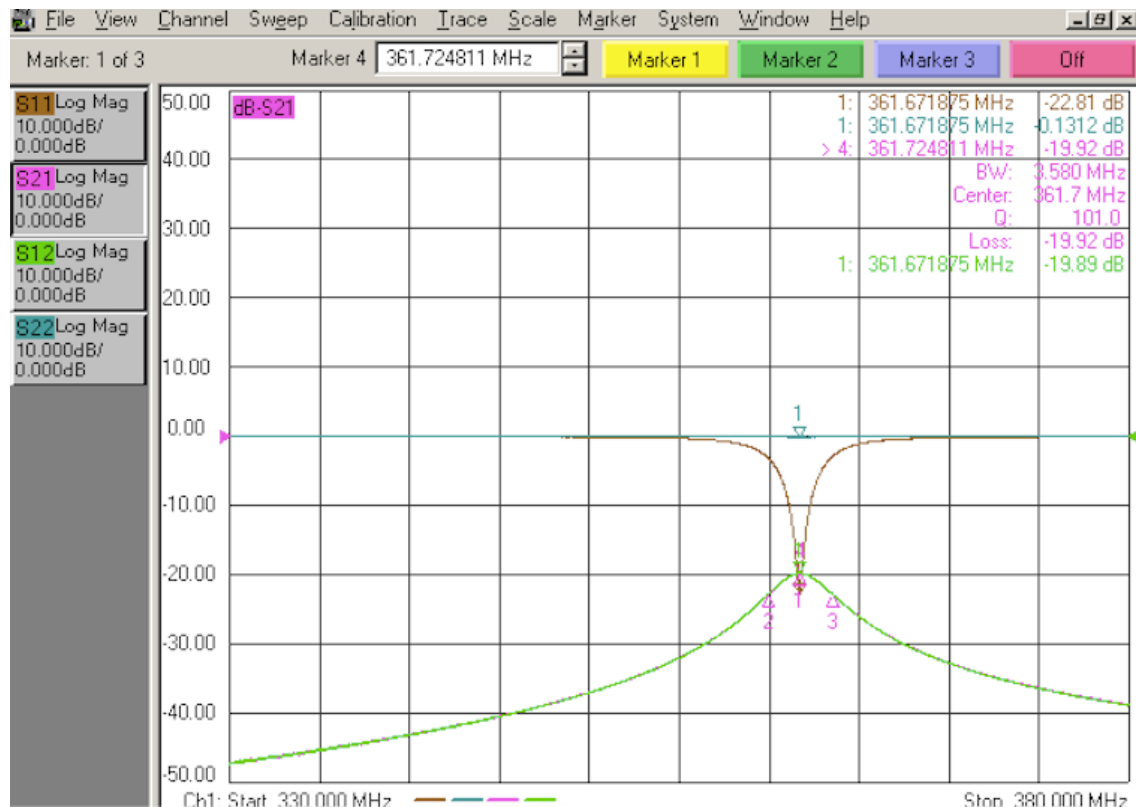


Figura 46. Medida del factor de calidad con el VNA.

Analizando la imagen anterior, se puede ver como en la traza del S_{21} (en rosa) el analizador de redes ha calculado el factor de calidad utilizando el método de los 3 puntos, quedando una Q cargada de 101. Antes de realizar la medida con el programa realizado en Python, se representó tanto el S_{11} como el S_{22} en la carta de Smith como se muestra en la imagen a continuación.

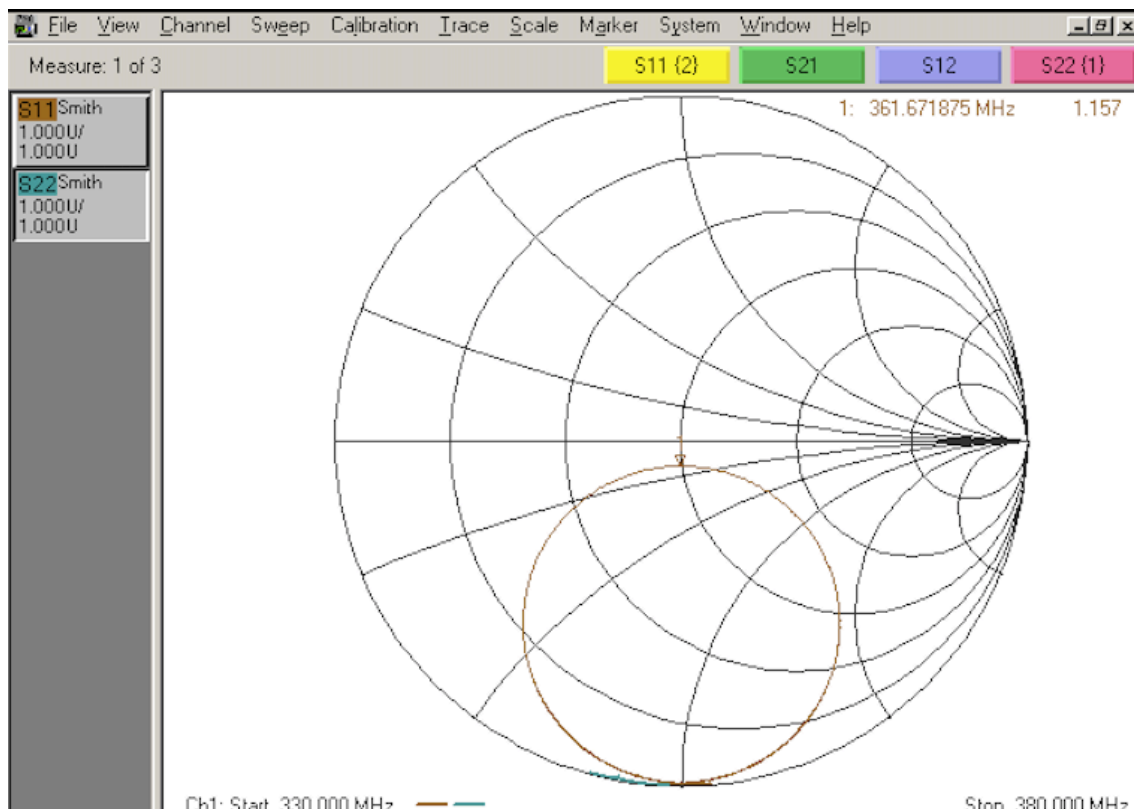


Figura 47. Representación S_{11} y S_{22} en carta de Smith.

Como se puede observar en la figura anterior, el S_{22} no va a presentar el tamaño suficiente como para que el programa de Python pueda realizar una medida en reflexión desde el puerto 2.

A continuación, se realiza la simulación del programa y en la siguiente figura se muestra el resultado óptimo que mostraría la interfaz de usuario tras realizar los cálculos del factor de calidad.

Figura 48. Medida del factor de calidad del resonador a 361 MHz.

Como se puede observar presenta una Q cargada de 100, mientras que el método de los tres puntos utilizado con el analizador de redes aproximaba una Q de 101, por tanto, se puede concluir que aparentemente el programa funciona correctamente. En la figura siguiente se presentan todos los cálculos que ha realizado el programa antes de presentar por pantalla el resultado óptimo del factor de calidad.

Los resultados en reflexion, para el puerto 1, son los siguientes:

Método de los 3 puntos:

$f_0 = 361.671875$ MHz
 $BW = 3615.439$ KHz
 $QL = 100.0$
 $\beta_{a1} = 0.8649$
 $\beta_{a2} = 0$
 $Qu = 187.0$

Método de kajfez:

$f_0 = 361.67292$ MHz
 $BW = 3453.799$ KHz
 $QL = 105.0$
 $\beta_{a1} = 0.8613$
 $\beta_{a2} = 0$
 $Qu = 195.0$

No se pueden dar valores para reflexión desde el puerto 2

No se puede realizar la corrección para los valores de los dos puertos

Los resultados en transmision son los siguientes:

Métodos de los 3 puntos:

$f_0 = 361.6875$ MHz
 $BW = 3587.099$ KHz
 $QL = 101.0$
 $\beta_{a1} = 0.877$
 $\beta_{a2} = 0.0146$
 $Qu = 191.0$

Métodos de K. Leong:

$f_0 = 361.703125$ MHz
 $BW = 3587.254$ KHz
 $QL = 100.0$
 $\beta_{a1} = 0.8613$
 $\beta_{a2} = 0.0$
 $Qu = 187.0$

Figura 49. Cálculos del factor de calidad del resonador a 361 MHz.

Analizando la figura anterior, se puede observar que la medida en reflexión del puerto 2 el programa no la ha realizado como se podía predecir en la figura 48. Esto ha ocurrido porque la diferencia que presenta el máximo del S_{22} menos el mínimo del S_{22} no supera el umbral comentado con anterioridad, lo que implica que el resonador presenta un S_{22} muy pequeño y la medida para el puerto 2 no se realiza para evitar resultados erróneos.

Tampoco se ha realizado la siguiente medida que aplicaba el factor de corrección, esto ha sido así porque para aplicar el factor de corrección tiene que haber dos factores de acoplo no despreciables y en este caso el factor de acoplo para el puerto 2 puede considerarse despreciable, por tanto, no se realiza la medida. Si se observa el resultado obtenido por K. Leong que es el considerado óptimo, la beta correspondiente al puerto 2 se ha supuesto 0 por la misma razón explicada anteriormente. En el método de Kajfez también se considera el factor de acoplo del puerto 2 despreciable.

La siguiente medida se realizó en las instalaciones de ESS Bilbao. Se efectuó la medida del factor de calidad en una cavidad Buncher, como la de la figura 4. Para realizar la medida se utilizó un VNA, un ordenador portátil donde se encontraba el programa realizado y el conversor GBIP-USB. El analizador de redes se conectó a la cavidad Buncher de manera que el puerto 1 se conectó al acoplador de potencia en la parte superior de la cavidad y el puerto 2 a la sonda o pickup en la zona lateral. En la imagen a continuación se puede observar la conexión de los puertos al analizador.



Figura 50. Conexión de los puertos al analizador.

En primer lugar, se realizó una medida en la que la β_1 era prácticamente 1 y la β_2 era prácticamente 0. Para conseguir aproximadamente esos factores de acoplo se giró el lazo de acoplo magnético del acoplador de potencia que aparece en la parte superior de la cavidad Buncher en la figura anterior, intentando que el S_{11} pasará por el centro de la carta de Smith. Igualmente, se giró el lazo de la sonda “pickup” para un acoplo mínimo. En la figura a continuación se muestra la representación de los diferentes parámetros S que aparece en el analizador de redes y se puede observar como el factor de acoplo del puerto 1 es muy cercano al 1.

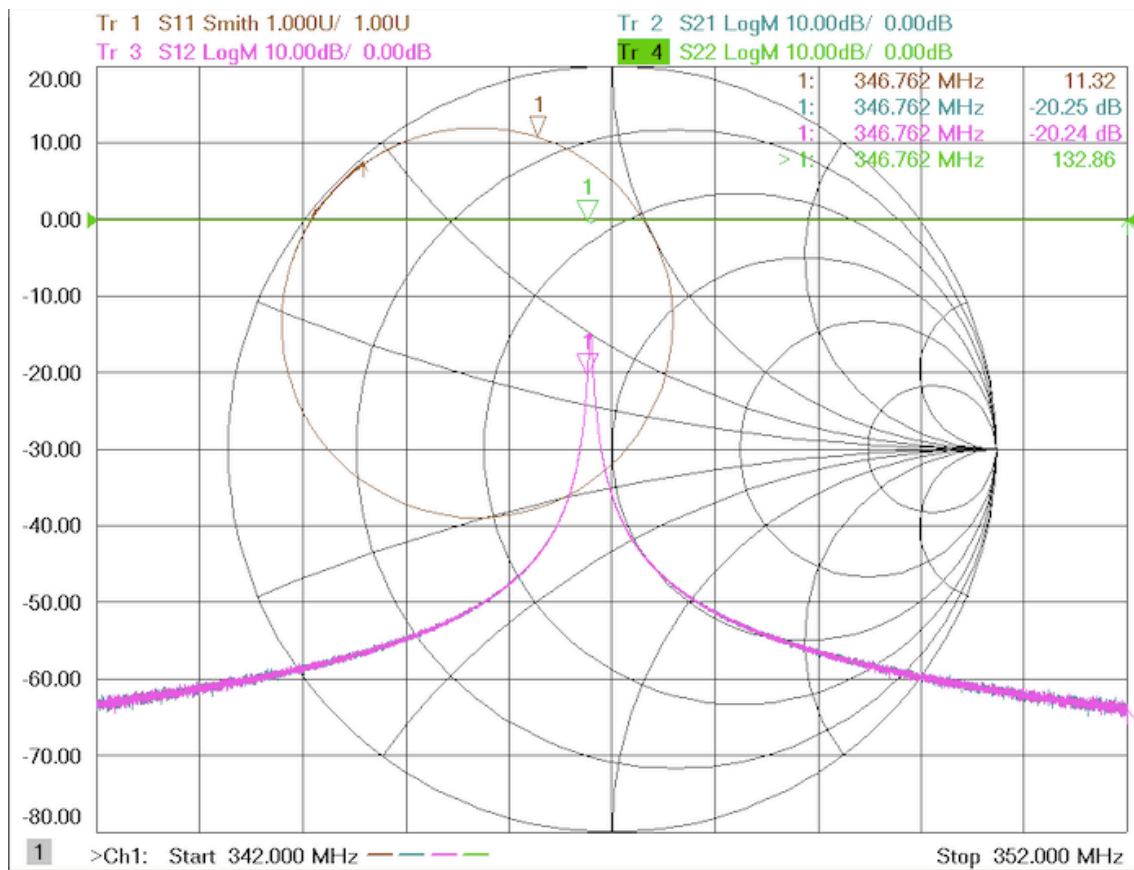


Figura 51. Cavity Buncher con factor de acoplo 1 y 0.

Teniendo en cuenta que es una cavidad resonante que se va a utilizar en un acelerador de partículas el factor de calidad que va a presentar va a ser muy alto. A continuación, se muestra los resultados obtenidos al ejecutar el programa para una cavidad de estas características.

ESS BILBAO: QUALITY FACTOR MEASUREMENT

☐ VNA
☒ Input File

File: prueba b1 1 b2 0 8_51.s2p

Browse

☐ Show Results
☐ History
☐ Verbose

Calculate

Output Data
 f0 (MHz): 346,79194
 Qo: 20031
 Ql: 9982
 beta1: 1,0066
 beta2: 0

Figura 52. Resultado óptimo del factor de calidad mostrado en la interfaz de usuario.

Como se puede observar en la figura, la β_1 presenta un valor de prácticamente 1 y la β_2 presenta un valor de 0 que es lo que se estaba buscando al mover la pieza de cobre mencionada. El factor de calidad descargado que el programa considera óptimo tiene un valor muy cercano a los 20000, lo cual tiene sentido, como se ha comentado anteriormente las cavidades Buncher presentan un factor de calidad muy alto. En la siguiente figura se muestran todos los factores de calidad que el programa ha calculado además del que ha considerado óptimo.

Los resultados en reflexion, para el puerto 1, son los siguientes:

Método de los 3 puntos:

f0 = 346.7925 MHz
BW = 34.66 KHz
QL = 10006.0
beta1 = 1.0728
beta2 = 0
Qu = 20739.0

Método de kajfez:

f0 = 346.79194 MHz
BW = 34.679 KHz
QL = 10000.0
beta1 = 1.0066
beta2 = 0
Qu = 20066.0

No se pueden dar valores para reflexión desde el puerto 2

No se puede realizar la corrección para los valores de los dos puertos

Los resultados en transmision son los siguientes:

Métodos de los 3 puntos:

f0= 346.79125 MHz
BW= 34.801 KHz
QL= 9965.0
beta1= 1.086
beta2= 0.0059
Qu= 20846.0

Métodos de K. Leong:

f0= 346.79194 MHz
BW= 34.801 KHz
QL= 9982.0
beta1= 1.0066
beta2= 0.0
Qu= 20031.0

Figura 53. Resultados para la primera medida de la cavidad Buncher.

Como era de esperar, solo se han realizado medidas en reflexión en el puerto 1 y en transmisión. Todas las betas calculadas coinciden con el valor esperado al principio de la simulación, es decir, las β_1 entorno a 1 y las β_2 entorno al 0. No se han realizado las medidas en el puerto 2 debido a que solo hay un factor de acoplo no despreciable, por tanto, las medidas del puerto 2 no se realizan. Además, como se puede observar todas las Q cargadas tienen un valor muy cercano a los 10000. Aparentemente el programa ha realizado la medida del factor de calidad correctamente.

La siguiente medida de la cavidad Buncher que se realizó fue para una β_1 mayor que 1 y una β_2 algo mayor que 0. De nuevo, para internar ajustar las betas al valor deseado se giró el acoplador de potencia de la parte superior de la cavidad y la sonda “pickup”. A continuación, se muestra lo que se observaba en la pantalla del analizador de redes antes de realizar la medida del factor de calidad con el programa en Python.

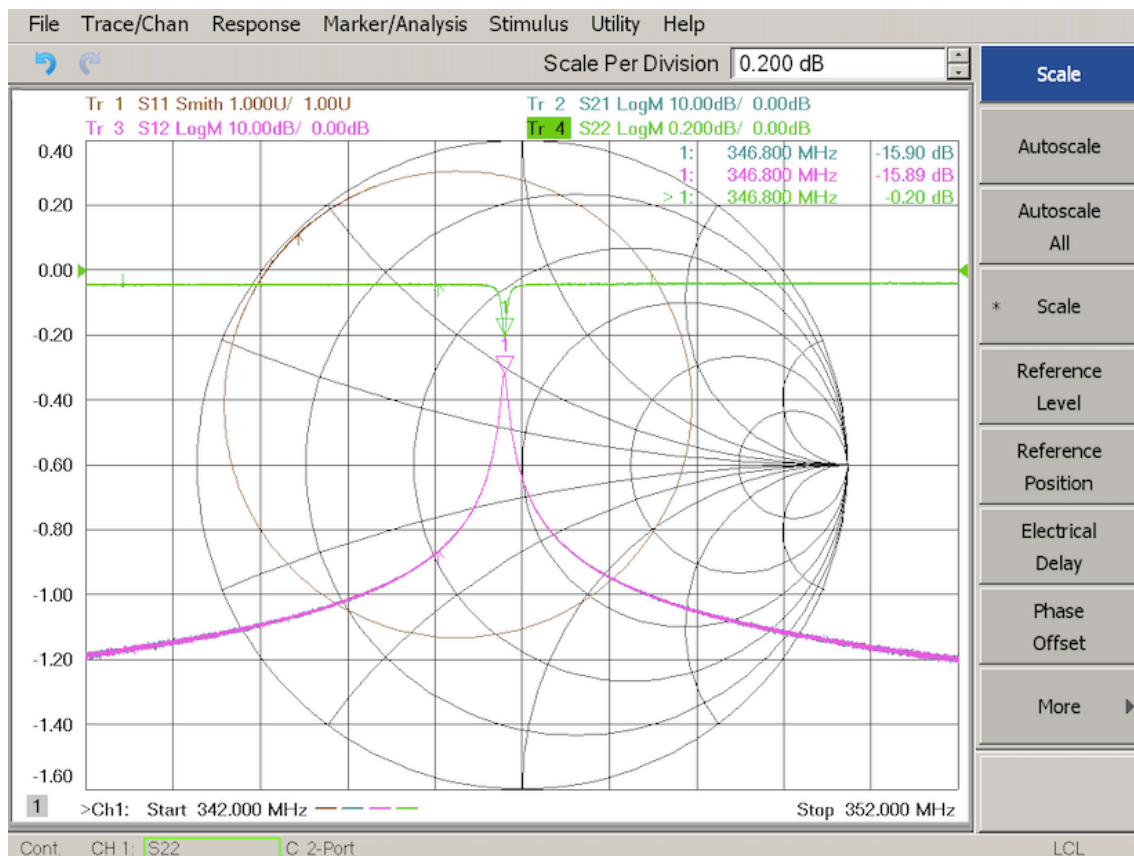


Figura 54. Medida cavidad Buncher β_1 mayor que 1 y β_2 mayor que 0.

Como se puede observar en esta imagen, aparentemente el S_{11} está sobre acoplado, por tanto, cuando se realice la medida con el programa quedara una β_1 bastante mayor que 1. A continuación, se muestra el resultado óptimo obtenido al simular el programa para los parámetros S que aparecen en la figura anterior.

ESS BILBAO: QUALITY FACTOR MEASUREMENT		UC
Input Data <input type="radio"/> VNA <input checked="" type="radio"/> Input File	File: Prueba b1 26 b2001 9_17.s2p <input type="button" value="Browse"/>	
<input type="checkbox"/> Show Results <input type="checkbox"/> History <input type="checkbox"/> Verbose	<input type="button" value="Calculate"/>	Output Data f0 (MHz): 346,794853 Qo: 19206 Ql: 5403 beta1: 2,5549 beta2: 0

Figura 55. Resultado óptimo de la cavidad Buncher sobre acoplada.

Observando la figura anterior, la β_1 presenta un valor de 2,6 aproximadamente y la β_2 entorno al 0, que es lo que se estaba buscando al ajustar el acoplador de la cavidad. Además, se puede observar como el valor de Q cargada ha reducido su valor a la mitad, pasando de tener un factor de calidad de aproximadamente 10000 a presentar un factor de calidad cargado de entorno los 5000. Mientras que el valor de Q descargada se mantiene constante respecto a las simulaciones anteriores, esto ocurre porque es el valor de Q intrínseca de la cavidad. En la siguiente figura al igual que se ha hecho en las medidas anteriores, se puede observar los resultados medidos tanto en reflexión como en

transmisión para el puerto 1, ya que el puerto 2 no presenta un factor de acoplo y se puede despreciar.

Los resultados en reflexion, para el puerto 1, son los siguientes:

Método de los 3 puntos:

f0 = 346.795 MHz
BW = 64.179 KHz
QL = 5404.0
beta1 = 2.6129
beta2 = 0
Qu = 19522.0

Método de kajfez:

f0 = 346.794853 MHz
BW = 64.031 KHz
QL = 5416.0
beta1 = 2.5549
beta2 = 0
Qu = 19254.0

No se pueden dar valores para reflexión desde el puerto 2

No se puede realizar la corrección para los valores de los dos puertos

Los resultados en transmision son los siguientes:

Métodos de los 3 puntos:

f0= 346.795 MHz
BW= 64.224 KHz
QL= 5400.0
beta1= 2.7277
beta2= 0.0439
Qu= 20366.0

Métodos de K. Leong:

f0= 346.794853 MHz
BW= 64.224 KHz
QL= 5403.0
beta1= 2.5549
beta2= 0.0
Qu= 19206.0

Figura 56. Todos los resultados obtenidos por el programa para una β_1 mucho mayor que 1

Para finalizar las medidas con la cavidad Buncher, se realizó una tercera y última medida del factor de calidad. En esta medida se intentó que el factor de acoplo del puerto 1 fuese menor que 1. En la figura a continuación se muestra el S_{11} infra acoplado que se observaba en el analizador de redes.

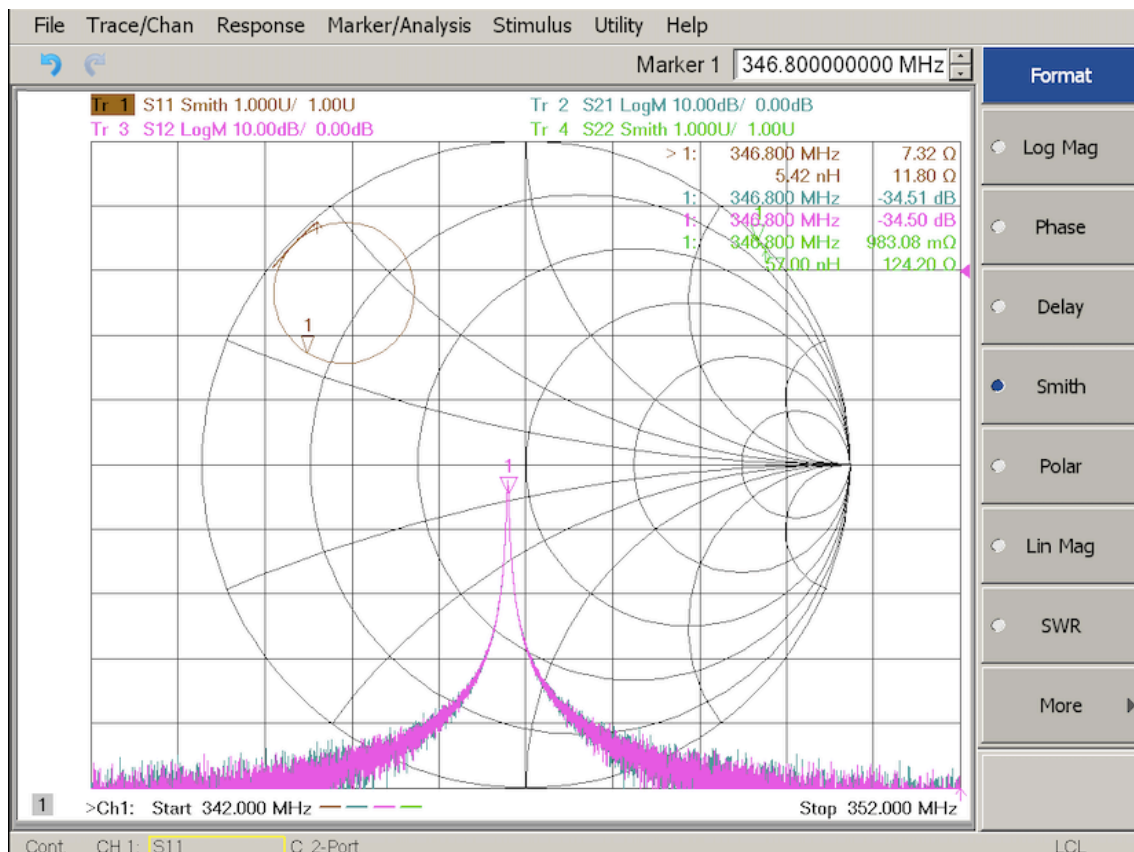


Figura 57. Cavity Buncher con una β_1 menor que 1

Una vez que el puerto 1 presenta las características que se estaba buscando para realizar una tercera medida distinta a las anteriores, se puso en marcha el programa de medida de factor de calidad, quedando como resultado óptimo el que se muestra en la interfaz de usuario de la figura siguiente.

ESS BILBAO: QUALITY FACTOR MEASUREMENT

Input Data: ☐ VNA ☒ Input File

File: prueba b1 028 b2 0 9.s2p

Calculate

Output Data:

f0 (MHz):	346,791088
Qo:	20112
Ql:	15730
beta1:	0,2786
beta2:	0

Figura 58. Representación resultados en la interfaz de usuario para una β_1 menor que 1.

Analizando tanto los resultados óptimos de la interfaz de usuario como el resto de los resultados del programa, se puede observar que la β_1 es mucho menor que uno, y la β_2 es de 0. Se obtiene un factor de calidad cargado de aproximadamente 15000, es el factor de calidad cargado más alto que se ha conseguido alcanzar de todas las medidas que se realizaron, mientras que el factor de calidad descargado en todas las medidas está entorno a 1 20000, siendo el de esta tercera medida el que mejor Q descargado presenta.

Los resultados en reflexion, para el puerto 1, son los siguientes:

Método de los 3 puntos:

f0 = 346.79125 MHz
BW = 22.252 KHz
QL = 15585.0
beta1 = 0.2852
beta2 = 0
Qu = 20029.0

Método de kajfez:

f0 = 346.791088 MHz
BW = 21.99 KHz
QL = 15770.0
beta1 = 0.2786
beta2 = 0
Qu = 20164.0

No se pueden dar valores para reflexión desde el puerto 2

No se puede realizar la corrección para los valores de los dos puertos

Los resultados en transmision son los siguientes:

Métodos de los 3 puntos:

f0= 346.79125 MHz
BW= 22.324 KHz
QL= 15534.0
beta1= 0.2863
beta2= 0.004
Qu= 20045.0

Métodos de K. Leong:

f0= 346.791088 MHz
BW= 22.324 KHz
QL= 15730.0
beta1= 0.2786
beta2= 0.0
Qu= 20112.0

Figura 59. Resultados de los factores de calidad para una β_1 menor que 1

La conclusión que se ha obtenido tras realizar las medidas de la cavidad Buncher es que cuanto mayor sea el factor de acoplo menor factor de calidad cargado presenta, esto ocurre debido a que el valor de Q_L tiene una relación inversamente proporcional con los acoplos, por tanto, cuanto mayor sea el valor de este menor será el valor el factor de calidad cargado. Todo lo contrario, ocurre cuando los acoplos son menores que uno, el factor de calidad cargado será mayor. En la expresión que se muestra a continuación se ve descrita la relación entre los acoplos y el factor de calidad cargado.

$$Q_L = \frac{Q_u}{1+\beta_2+\beta_2} \quad (104)$$

Además, como se puede ver a partir de los últimos resultados descritos y como predice la fórmula (104), el factor de calidad descargado (Q_u) es aproximadamente constante (cercano a 20000). Esto se puede comprobar que todas las Q_u calculadas con anterioridad correspondientes a la cavidad Buncher. Estos resultados están de acuerdo con las estimaciones realizadas anteriormente por ESS Bilbao.

Asimismo, se pudo comprobar que los resultados de factor Q y factores de acoplo ofrecidos por los métodos de 3 puntos presentan una mayor variabilidad al ejecutarlos en repetidas ocasiones. En cambio, los métodos de Kajfez y Leong proporcionan resultados consistentes y repetitivos.

Capítulo 5: Conclusión y líneas futuras

En este trabajo de fin de máster, es el resultado de una colaboración entre el departamento de ingeniería de comunicaciones de la Universidad de cántabra y el consorcio ESS Bilbao. El nombre que recibe el proyecto que llevado a cabo es: “desarrollo de software de medida avanzada de factor de calidad en cavidades resonantes para aceleradores de partículas”.

La idea de realizar este TFM surge ante la necesidad de un alto factor de calidad en las cavidades resonantes que forman parte de un acelerador de partículas, ya que esto implica una baja tasa de pérdidas de energía en relación con la energía “útil” almacenada en el propio resonador.

Para realizar el programa encargado de realizar las medidas del factor de calidad, se ha escogido un lenguaje de programación de software libre, Python. Lo que ha supuesto una dificultad añadida, debido a que era la primera vez que trabajaba con este lenguaje de programación.

Previamente a la realización del programa se ha realizado un análisis de la biografía técnica relevante. La lectura realizada incluye los diferentes métodos matemáticos que se han utilizado durante el programa para realizar el cálculo del factor de calidad, en reflexión y en transmisión, como la teoría introductoria a los aceleradores de partículas y sus cavidades resonantes.

Los datos utilizados por el software para realizar el cálculo del factor de calidad de la cavidad resonante se han obtenido a través de un analizador de redes, por tanto, en el TFM además de realizar un programa de medida de factor de calidad, se ha controlado tanto manualmente como remotamente instrumentos de medida de radiofrecuencia especializados, con el objetivo de realizar medidas avanzadas de caracterización de cavidades resonantes para aceleradores de partículas

Una vez realizado el programa, se ha comprobado el correcto funcionamiento de éste realizando múltiples simulaciones con la cavidad Buncher de ESS Bilbao, cada una de éstas con diferentes valores de los factores de acoplo. A partir del análisis de los resultados obtenidos a partir del software se ha concluido que el factor de calidad cargado depende directamente del factor de acoplo, como se ha podido observar al realizar el análisis teórico previo, mientras que el factor de calidad descargado no ha variado su valor prácticamente en las diferentes simulaciones. Así se ha podido comprobar que este valor es el intrínseco de la cavidad, en este caso, el de la cavidad Buncher. Además, analizando más profundamente los resultados obtenidos, se ha observado cómo a medida que el factor de acoplo aumenta, el factor de calidad cargado que presenta es menor. Esto ocurre debido a que éste tiene una relación inversamente proporcional con los acoplos. Por tanto, se ha concluido que para obtener un factor de calidad cargado alto el factor de acoplo tiene que ser lo más pequeño posible.

El software realizado ha permitido la medida del factor de calidad en resonadores cuya medida se realiza de manera convencional, es decir, en reflexión o en transmisión.

Como líneas futuras se propone añadir una función nueva en el software de medida de factor de calidad que contemple la caracterización de resonadores en medidas de tipo reacción. Esta nueva función serviría para realizar la caracterización de resonadores dieléctricos. Este tipo de resonadores de altísimo factor de calidad suelen emplearse acoplados a una línea microstrip y al medir en reflexión presentan un círculo Q que no es tangente al borde de la carta de Smith sino que lo es siempre al centro (independientemente del valor del acoplo). Por tanto, los métodos presentados en este TFM no sirven tal cual para caracterizarlos.

Bibliografía

- [1] Dotson, B. (2014, 18 junio). *How Particle Accelerators Work*. Energy.Gov.
<https://www.energy.gov/articles/how-particle-accelerators-work>
- [2] Cárdenas, R. G. (2020). *Tipos de aceleradores de partículas*. La naturaleza de la materia. http://www.cienciorama.unam.mx/a/pdf/235_cienciorama.pdf
- [3] Xabier Cid Vidal. (2010). *Acercándonos al LHC - Acelerador de partículas*. Acercándose al LHC. https://www.lhc-closer.es/taking_a_closer_look_at_lhc/0.particles_accelerator/idioma/es_ES
- [4] de la Nuez, D. (2018, 2 abril). *7 importantes descubrimientos realizados por el CERN*. VIX. <https://www.vix.com/es/btg/curiosidades/7110/7-importantes-descubrimientos-realizados-por-el-cern>
- [5] *ESS Bilbao – Strategic hub for neutron science and technologies*. (2010). ESS Bilbao. <https://www.essbilbao.org/es/>
- [6] Amaldi, U. Á. F. (2011, 4 julio). *Pequeños aceleradores de partículas para aplicaciones médicas*. Revista Método. <https://metode.es/revistas-metode/monograficos/pequenos-aceleradores-de-particulas-para-aplicaciones-medicas.html>
- [7] Barranco Garcia, J. (2010). *Introducción a la física de aceleradores*. *Introducción a la física de aceleradores*. Published.
https://indico.cern.ch/event/99174/contributions/1292435/attachments/1120684/1599189/Intro_aceleradores_particulas_28_06_10_HST.pdf
- [8] Ministerio de ciencia e innovación. (2016). *Cavidad Buncher*. Ciemat.
<https://www.ciemat.es/cargarAplicacionNoticias.do;jsessionid=8E6ACEB8C5672FDD2759459CC4D4A2CB?idArea=0&identificador=1038>

- [9] CERN. The Radio Frequency Quadrupole (RFQ). *CERN*. Published.
<https://cds.cern.ch/record/1005049/files/p201.pdf>
- [10] Coursera. (s. f.). *Drift tube linac (DTL)*. Recuperado 6 de septiembre de 2021, de
<https://www.coursera.org/lecture/fundamentals-particle-accelerator-technology/drift-tube-linac-dtl-VzgTq>
- [11] Byrd, J. (2009, junio). *Resonant Cavities*. University of New Mexico.
- [12] Kajfez, D. *Q-factor*. Encyclopdia RF Microwave Qfactor.
- [13] Nagore Garmendia, & Pedro González. *Measurement of Quality Factor and Coupling Factor of Cavities*. ESS Bilbao.
- [14] Kajfez, D. (1994, julio). *Linear Fractional Curve Fitting for Measurement of High Q Factor*.
- [15] Leong, K. (2002, septiembre). *Precise Measurements of the Q Factor of Dielectric Resonators in the Transmission Mode—Accounting for Noise, Crosstalk, Delay of Uncalibrated Lines, Coupling Loss, and Coupling Reactance*.
- [16] Álvaro Guitián Herrera. (2020). *Contribución al proceso de reacondicionamiento de amplificadores klystron para aceleradores lineales de partículas en ESS Bilbao*. Trabajo fin de Máster.
- [17] *RO4003C™ Laminates - Rogers Corporation*. (s. f.). Datasheet.
<https://rogerscorp.com/advanced-electronics-solutions/ro4000-series-laminates/ro4003c-laminates>

ANEXO: Código fuente

```
#####
##### CALCULO DEL FACTOR DE CALIDAD (Q FACTOR) #####
#####

"""
Created on Mo Feb 23 11:04:00 2021

@author: Edurne Monteoliva y Alejandro Gutierrez
"""

# Import Libraries
import numpy as np
import matplotlib
import math
from matplotlib import pyplot as plt
matplotlib.rcParams.defaults()
%matplotlib notebook

#####
#Lectura de fichero S2P
def lees2p(filename):
    # Function that reads a .s2p file, i.e. an ascii file according to the so called
    # "Touchstone" format, Listing the scattering parameters (S parameters) of a
    # 2-port network as a function of frequency.
    # frequency and 4 complex valued S parameters (freq, S11, S21, S12 and S22) are
    # stored in 9 columns

    if filename.lower().split('.')[1] != 's2p':
        print('ERROR: illegal file extension', filename[-3:-1]) #error si no es s2p

    with open(filename, "r") as files2p:
        f,S11_1,S11_2,S21_1,S21_2,S12_1,S12_2,S22_1,S22_2 = [],[],[],[],[],[],[],[],[
]
        #genera tantos arrays como parametros tiene el fichero

        while 1:
            line = files2p.readline()
            #print(line)

            if not type(line) == str:
                line = line.decode("ascii")
            if not line: # end function
                break
            if len(line) == 0: # continue
                continue
            if line[0] == "!": # comments
                continue
            if line[0] == "#": # this means options
                options = line[1:].strip().split()
                funits = options[0].lower()
                tipoparam = options[1].lower()
                formato = options[2].lower()
                Z0 = options[4]

                if funits not in ['hz', 'khz', 'mhz', 'ghz']:
                    print('ERROR: illegal frequency_unit %s', funits)
                    fmult = {'hz': 1, 'khz': 1e3, 'mhz': 1e6, 'ghz': 1e9}.get(funits)

                if tipoparam not in 's': # originally 'syzgh'
                    print('ERROR: illegal parameter value %s', tipoparam)

                if formato not in ['ma', 'db', 'ri']:
```



```

        print('ERROR: illegal format value %s]' % formato)

    else: # read the parameters
        f.append(fmult*float(line.split()[0]))
        S11_1.append(float(line.split()[1]));
        S11_2.append(float(line.split()[2]));
        S21_1.append(float(line.split()[3]));
        S21_2.append(float(line.split()[4]));
        S12_1.append(float(line.split()[5]));
        S12_2.append(float(line.split()[6]));
        S22_1.append(float(line.split()[7]));
        S22_2.append(float(line.split()[8]));

f = np.array(f)
S11_1 = np.array(S11_1); S11_2 = np.array(S11_2)
S21_1 = np.array(S21_1); S21_2 = np.array(S21_2)
S12_1 = np.array(S12_1); S12_2 = np.array(S12_2)
S22_1 = np.array(S22_1); S22_2 = np.array(S22_2)
#print('S11_1', S11_1)

# Complex S parameters calculated depending on the format specified in s2p file
if formato == 'ri':
    S11 = S11_1 + 1j*S11_2;    S21 = S21_1 + 1j*S21_2
    S12 = S12_1 + 1j*S12_2;    S22 = S22_1 + 1j*S22_2
if formato == 'ma':
    S11 = S11_1 * np.exp(1j*S11_2*np.pi/180)
    S21 = S21_1 * np.exp(1j*S21_2*np.pi/180)
    S12 = S12_1 * np.exp(1j*S12_2*np.pi/180)
    S22 = S22_1 * np.exp(1j*S22_2*np.pi/180)
if formato == 'db':
    S11 = 10**((S11_1/20) * np.exp(1j*S11_2*np.pi/180)
    S21 = 10**((S21_1/20) * np.exp(1j*S21_2*np.pi/180)
    S12 = 10**((S12_1/20) * np.exp(1j*S12_2*np.pi/180)
    S22 = 10**((S22_1/20) * np.exp(1j*S22_2*np.pi/180)
#print('S11', S11)

return f, S11, S21, S12, S22
#####

#####

def Call_File(fichero):

    filename = fichero.files
    #filename = 'B_Coupler_Pickup_19000_1.1_0.0235_0_0.s2p'

    f, S11, S21, S12, S22 = lees2p(filename)

    #pasa los parametros S a magnitud db y angulo
    f_MHz = f/1e6
    MS11 = np.absolute(S11);    MS21 = np.absolute(S21)
    MS12 = np.absolute(S12);    MS22 = np.absolute(S22)
    dBS11 = 20*np.log10(MS11);    dBS21 = 20*np.log10(MS21)
    dBS12 = 20*np.log10(MS12);    dBS22 = 20*np.log10(MS22)
    AS11 = np.angle(S11)*180/np.pi;    AS21 = np.angle(S21)*180/np.pi
    AS12 = np.angle(S12)*180/np.pi;    AS22 = np.angle(S22)*180/np.pi

    #dibuja los parametros S

    #print()
    #print('freq (MHz): '+str(f_MHz))
    #print()
    #print('S11: '+str(S11))
    #print('dBS11: '+str(dBS11))
    #print('AS11: '+str(AS11))

```

```

#print()
#print('re(S11): '+str(np.real(S11)))
#print('im(S11): '+str(np.imag(S11)))
#print()

#from matplotlib import pyplot as plt
#%matplotlib notebook

if Results:
    fig = plt.figure(figsize=(8.5,4.5))
    ax1 = fig.add_subplot(111)
    ax1.set_title('S Parameters') # figure title
    ax1.set_xlabel('f [MHz]') # x axis label
    ax1.set_ylabel('MAG S [dB]', color='blue') # left y axis label
    ax1.set_xlim(left = min(f_MHz), right = max(f_MHz)) # initial x limits
    ax1.set_ylim(-80, 10) # y limits
    ax1.tick_params(axis='y', labelcolor='blue') # ticks
    ax1.axes.grid('both') #grid

    ax2 = ax1.twinx() # duplicate properties to right y axis
    ax2.set_ylabel('ANGLE S [deg]', color='red') # right y axis label
    ax2.set_ylim(-200, 200) # y limits
    ax2.tick_params(axis='y', labelcolor='red') # ticks

    line11, = ax1.plot(f_MHz, dBS11, 'y', label='S11')
    line21, = ax1.plot(f_MHz, dBS21, 'b', label='S21')
    line21, = ax1.plot(f_MHz, dBS12, 'm', label='S21')
    line22, = ax1.plot(f_MHz, dBS22, 'g', label='S22')
    line22, = ax2.plot(f_MHz, AS21, 'r', label='S21ph')

    fig.legend() #(loc='center', bbox_to_anchor=(0.07, 0.7))
    fig.tight_layout() # otherwise the right y-label is slightly clipped

    fig.show()
    #fig.canvas.draw()

    return f, S11, S21, S12, S22
#####

#####

def PNA():
    # https://pyvisa.readthedocs.io/en/latest/introduction/getting.html
    # Open Anaconda Prompt:
    # pip install -U pyvisa
    # pip install -U pyvisa-py

    # usb-gpib converter (Agilent, NI,...)
    # Agilent/Keysight I/O libraries, NI MAX (measurement automation explorer)

    # Access the PNA N5241A network analyzer via NI USB-GPIB ADAPTER
    # connect the GPIB to Talker/Listener GPIB(0) connector

    import numpy as np
    import pyvisa
    pyvisa.ResourceManager('@py')

    rm = pyvisa.ResourceManager()
    print(rm)
    print(rm.list_resources())

    pna = rm.open_resource('GPIB0::16::INSTR')
    print(pna)
    pna.timeout = 5000

```

```

#pna.read_termination = '\n'
#pna.write_termination = '\n'

print(pna.query('*IDN?'))

pna.write('CALC:PAR:DEF "Meas1",S11')
pna.write('CALC:PAR:DEF "Meas2",S21')
pna.write('CALC:PAR:DEF "Meas3",S12')
pna.write('CALC:PAR:DEF "Meas4",S22')

pna.write('disp:wind1:trac1:feed "Meas1"')
pna.write('disp:wind1:trac2:feed "Meas2"')
pna.write('disp:wind1:trac3:feed "Meas3"')
pna.write('disp:wind1:trac4:feed "Meas4"')

print(pna.query('CALC:PAR:CATalog?'))

# Para saber si está calibrado o no
pna.write('CALC:PAR:SEL "Meas1"')
cal1 = pna.query('CALC:CORR:STATE?')
print(cal1)
pna.write('CALC:PAR:SEL "Meas1"')
cal1 = pna.query('CALC:CORR:STATE?')
pna.write('CALC:PAR:SEL "Meas2"')
cal2 = pna.query('CALC:CORR:STATE?')
pna.write('CALC:PAR:SEL "Meas3"')
cal3 = pna.query('CALC:CORR:STATE?')
pna.write('CALC:PAR:SEL "Meas4"')
cal4 = pna.query('CALC:CORR:STATE?')

print(cal1,cal2,cal3,cal4)
#if cal1 == 'NONE' or cal2 == 'NONE' or cal4 == 'NONE':
#    print('VNA no calibrado')
#    break

# Set byte order to swapped (Little-endian) format
pna.write('FORM:BORD SWAP')
# Set data type to real 64 bit binary block
pna.write('FORM REAL,64')

# Read S2P data from PNA.
data = pna.query_binary_values('CALC:DATA:SNP?',datatype='d',is_big_endian=False)
#print(data)

# WITH ARRAYS
data = np.array(data)

# Reshape data so it is split into 9 columns (freq, S11dB, S11deg, S21...
N = int(len(data)/9)
print(N)
f = data[0:N]
dBS11 = data[N:2*N]; angS11 = data[2*N:3*N]
dBS21 = data[3*N:4*N]; angS21 = data[4*N:5*N]
dBS12 = data[5*N:6*N]; angS12 = data[6*N:7*N]
dBS22 = data[7*N:8*N]; angS22 = data[8*N:9*N]
#print(f, '\n',dBS11, '\n',angS11, '\n',dBS21, '\n',angS21, '\n',dBS12, '\n',angS12, '\n'
,dBS22, '\n',angS22, '\n')

f_MHz = f/1e6
S11 = 10**((dBS11/20) * np.exp(1j*angS11*np.pi/180)
S21 = 10**((dBS21/20) * np.exp(1j*angS21*np.pi/180)
S12 = 10**((dBS12/20) * np.exp(1j*angS12*np.pi/180)
S22 = 10**((dBS22/20) * np.exp(1j*angS22*np.pi/180)
#print(S11, '\n',S21, '\n',S12, '\n',S22)

from matplotlib import pyplot as plt

```

```

%matplotlib notebook

fig = plt.figure(figsize=(8,4))
ax1 = fig.add_subplot(111)
ax1.set_title('S Parameters') # figure title
ax1.set_xlabel('f (MHz)') # x axis label
ax1.set_ylabel('S (dB)', color='red') # left y axis label
ax1.set_xlim(left = min(f_MHz), right = max(f_MHz)) # initial x limits
ax1.set_ylim(-50, 10) # y limits
ax1.tick_params(axis='y', labelcolor='red') # ticks
line11, = ax1.plot(f_MHz, dBS11, 'y', label='S11')
line21, = ax1.plot(f_MHz, dBS21, 'b', label='S21')
line12, = ax1.plot(f_MHz, dBS12, 'm', label='S12')
line22, = ax1.plot(f_MHz, dBS22, 'g', label='S22')
fig.legend(loc='center left', bbox_to_anchor=(0.07, 0.7))
fig.tight_layout() # otherwise the right y-label is slightly clipped
ax1.axes.grid('both') #grid

fig.show()
fig.canvas.draw()

#pna.close() Mirar donde colocarlo

return f, S11, S21, S12, S22
#####

#####

def smithchart(rho1=0,line_style1='y-',label1='S11',rho2=0,line_style2='g-',
label2='None',ax=None):

    # Smith chart grid
    if ax == None:
        ax = plt.gca()
    ax.axis('equal')
    ax.axis('off')
    #fig.tight_layout() # otherwise the right y-label is slightly clipped
    fig.show()
    # Dibujo del circulo unidad
    thetas=np.linspace(0,2*np.pi,361)
    ax.plot(np.cos(thetas),np.sin(thetas),'k') # Circulo unidad

    # Dibujo de Los ejes (circulos R=cte, X=cte)
    rlis = np.array([0.2, 0.5, 1, 2, 5])
    xlis = np.array([0,0.2,0.5,1,2,5]) # only positive values

    for r in rlis:
        X = np.logspace(-3,2,101) # X ranges from 0.01 to 100
        Z = r + 1j*X
        rho = (Z-1)/(Z+1)
        ax.plot(np.real(rho),np.imag(rho),'r',lw=0.5,ls='--')
        ax.plot(np.real(rho),-np.imag(rho),'r',lw=0.5,ls='--')
        if r == 1:
            ax.plot(np.real(rho),np.imag(rho),'r',lw=1,ls='--')
            ax.plot(np.real(rho),-np.imag(rho),'r',lw=1,ls='--')

        ax.text((r-1)/(r+1),0.01,str(r))
    for x in xlis:
        R = np.logspace(-3,3,101) # R ranges from 0.001 to 1000
        Z = R + 1j*x
        rho = (Z-1)/(Z+1)
        ax.plot(np.real(rho),np.imag(rho),'r',lw=0.5,ls='--')
        ax.plot(np.real(rho),-np.imag(rho),'r',lw=0.5,ls='--')
        if x == 1:
            ax.plot(np.real(rho),np.imag(rho),'r',lw=1,ls='--')

```

```

        ax.plot(np.real(rho), -np.imag(rho), 'r', lw=1, ls='--')

    if x == 0:
        ax.text(-1.12, 0.0, str(x))
    else:
        ax.text((1.09*np.real(rho[0])-0.1), (1.05*np.imag(rho[0])), '+j'+str(x))
        ax.text((1.08*np.real(rho[0])-0.08), (-1.05*np.imag(rho[0])-0.05), '-j'+str
(x))

    # Dibujo de Los coeficientes de reflexión
    line1, = ax.plot(np.real(rho1), np.imag(rho1), line_style1, label=label1)

    if rho2 is not None:
        line2, = ax.plot(np.real(rho2), np.imag(rho2), line_style2, label=label2)

    if label1 is not None or label2 is not None:
        fig.legend()

    return
#####

#####
def Qu_refl(f, S11, inout, verbose):

    if verbose:
        print('REFLECTION MEASUREMENT');
        print('Metodo de 3 puntos (+/-45°) en la Carta de Smith')

    #pasa Los parametros S a magnitud db y angulo
    f_MHz = f/1e6
    MS11 = np.absolute(S11)
    dBS11 = 20*np.log10(MS11)
    AS11 = np.angle(S11)*180/np.pi

    # Busqueda de resonancia de S11 --> Llamar a funcion de busqueda de resonancia
    dBS11res, indres, S11res, f0, dBS11d, indd, S11d, fd = searchresonance(f, S11, dBS11, verbo
se)

    # Phase offset del círculo Q a la recta real
    S11off = phaseoffset(f, S11, verbose)

    # Buscamso puntos a +/-45° en S11 para QL, primero a Las frecuencias mas cercanas
    realS45 = (S11off.real[indres] + 1)/2 - 1;
    realS45 = (S11off.real[indres] + 1 + S11off.real[indd] + 1)/2 - 1;
    S11offd = np.roll(S11off, -1) # desplazamos Los elementos del array 1 posición hac
ia La izquierda
    LS11off = len(S11off)
    LS11offd = len(S11offd)

    #if verbose == 1:
        #print(S11offreal)
        #print(S11offdreal)
        #print(ind1)

    S11offreal = S11off.real
    S11offdreal = S11offd.real
    S11real = S11.real

    # Buscamos el indice que da justo por debajo de realS45 en el array original, y p
or tanto que en el array desplazado 1 posición da justo por encima de realS45
    ind1 = np.where( (S11offreal <= realS45) & (S11offdreal > realS45) )
    S11off[ind1]
    S11offd[ind1]
    ind1 = [x for x in ind1 if x < indres]

```

```

if verbose:
    print('#####')
    print('ind1: '+str(ind1))
    print('S11off[ind1]: '+str(S11off[ind1]))
    print('S11offd[ind1]: '+str(S11offd[ind1]))

# Notice that Logical-and is different from &
ind2 = np.where( (S11offreal >= realS45) & (S11offdreal < realS45) )
S11off[ind2]
#ind2 = ind2(np.where(ind2 > indres))
if verbose:
    print('ind2: '+str(ind2))
    print('S11off[ind2]: '+str(S11off[ind2]))
    print('S11offd[ind2]: '+str(S11offd[ind2]))

if not ind1: # Equivalente del isempty
    ind1 = 21;
    print('WARNING: No se encuentra real (S11) para 45 deg');

if not ind2:
    ind2 = size(f, axis(1));
    print('WARNING: No se encuentra real (S11) para 45 deg');

if len(ind1) > 1 or len(ind2) > 1:
    print('WARNING: Rizado alto. Considere aumentar el filtrado (Navg_pts)');

# Interpolamos
Af = f[1] - f[0]; # Ponemos un índice menor por Python
f1 = f[ind1[0]] + Af * (1 + ind1[-1] - ind1[0]) * (realS45 - S11offreal[ind1[0]])
/(S11real[ind1[-1]+1] - S11real[ind1[0]]);
f2 = f[ind2[0]] + Af * (1 + ind2[-1] - ind2[0]) * (realS45 - S11offreal[ind2[0]])
/(S11real[ind2[-1]+1] - S11real[ind2[0]]);

if verbose:
    print("f1: ",f1)
    print("f2: ",f2)
    print("Af: ",Af)
    print("f[ind1]: ",f[ind1])
    print("f[ind2]: ",f[ind2])

# Calculamos BW y QL
BW = abs(f2[0] - f1[0])
QL = f0/BW;

if verbose:
    print("BW: ",BW)
    print("QL: ",QL)

# Calculamos Qu
d = 1 + S11offreal[indres];
k = 1/(2/d - 1);
Qu = QL * (1 + k);

if verbose:
    print("k: ",k)
    print("Qu: ",Qu)
    print("d = ",d)

# Calculamos Qu con beta y Comprobamos si es under/overcoupled
undercoupled1 = checkundercoupling(f, S11,verbose)
VSWR1 = (1 + abs(S11[indres]))/(1 - abs(S11[indres]))

if undercoupled1 == 1:
    beta1 = 1/VSWR1
else:

```

```

    beta1 = VSWR1

Qu = QL * (1 + beta1)

results = [f0, BW, QL, beta1, 0, Qu]

#####
# "Linear fractional curve fitting for measurement of high Q factors", D.
# Kajfez, IEEE, Trans. Microwave Theory Tech. MTT-42:1149-1153(July 1994).

# Rhoi=(a1*t+a2)/(a3*t+1)
# t=2*(f-fL)/f0    normalized frequency variable
# a1*t(n) +a2 -a3*t(n)*Rhoi(n)=Rhoi(n),  n=1:Nf
# implicit form, linear combination of a1, a2, a3.
# Nf equations, 3 unknowns
# [A]*[x]=[b], [A]: rect. matrix (Nfx3), [x], [b]: column vectors
# [x] = [a1; a2; a3], [b]=Rhoi(:)
# Sol: [x]=[A]\[b]
# 2 Iterations are performed to improve fL evaluation
# the procedure is very sensitive to fL estimation error

for i in [1, 2]:

    f1 = f0
    # Restringimos el ancho de banda de evaluación
    spank = 3 * BW
    indk = np.where((f > (f1 - spank/2)) & (f < (f1 + spank/2)))
    fk = f[indk]
    S11k = S11[indk]
    t = 2 * (fk - f1)/f0

    # Old Method
    A = [t, np.ones(len(fk)), -t*S11k]
    At = np.transpose(A) # Transponemos La A para que cuadren Las dimensiones
    if verbose:
        print("Tamaño de A: ", np.shape(A))
        print("A = ", A)

    b = S11k

    x = np.linalg.lstsq(At, b,rcond=None)
    s = x[0]
    a1 = s[0]; a2 = s[1]; a3 = s[2]

    if verbose:
        print("Tamaño de b: ", np.shape(b))
        print("x = ", x)
        print("a1 = ", a1)
        print("a2 = ", a2)
        print("a3 = ", a3)

    # Calculamos Qu
    Rhos = a1/a3
    QL = a3.imag
    d = abs(a2-a1/a3)
    k = d/(2-d)
    Qu = QL*(1 + k)

    if verbose:
        print("Rhos = ", Rhos)
        print("QL = ", QL)
        print("d = ", d)
        print("k = ", k)
        print("Qu = ", Qu)

```

```

# Trazamos el círculo con muchos puntos
t = np.linspace(t[0], t[-1], 1601)
Rhoi = (a1*t+a2) / (a3*t+1)
#Se hace el cambio a dB porque la función min(complejo) opera sobre la parte
real
MSRhoi = np.absolute(Rhoi)
dBRhoi = 20*np.log10(MSRhoi)

# Volvemos a calcular la frecuencia de resonancia
S11resk, indresk = dBRhoi.min(0), dBRhoi.argmin(0) # Valor y posición
tresk = t[indresk]
fres = tresk*f0/2 + f0
f0 = fres

if verbose:
    print("tresk = ", tresk)
    print("fres = ", fres)
    print('#####')

d2 = 2;
if verbose:
    print(abs(Rhos))
if abs(Rhos) < 0.99:
    if verbose:
        print("Correction for coupling losses applied:")

    d = 2*abs((a2*a3-a1)/(a3-np.conj(a3)))
    if verbose:
        print("d: ", d)

#Cálculo del punto w_L de Kajfez
difRho = Rhoi - Rhos
MSdifRho = np.absolute(difRho)
dBdifRho = 20*np.log10(MSdifRho)
tmp, ind1 = dBdifRho.max(0), dBdifRho.argmax(0)
if verbose:
    print("ind1: ", ind1)
Rho1 = Rhoi[ind1]
if verbose:
    print("Rho1: ", Rho1)
# Calculate phi angle with generalised Pitagoras
argphi = -(abs(Rho1)**2-abs(Rhos)**2-d**2)/(2*abs(Rhos)*d)
if verbose:
    print("argphi: ", argphi)
if argphi > 1:
    argphi = 1
elif argphi < -1:
    argphi = -1
phi = math.acos(argphi)
if verbose:
    print("phi: ", phi)
# Calculate alpha angle internal sum of angles of a triangle
alpha = np.pi-phi-(abs(np.angle(Rhos)))
if verbose:
    print("alpha: ", alpha)

if abs(alpha) > np.pi/100:
    d2 = 2*abs(Rhos)*np.sin(abs(np.angle(Rhos)))/np.sin(abs(alpha))
else: # Si alpha es pequeño podrán surgir problemas numéricos
    d2 = 2*abs(Rhos)
if verbose:
    print("d2: ", d2)

if d2 > 2:
    d2 = 2
    print("WARNING: d2 resulta mayor de 2")

```



```

        if d2 < 0:
            d2 = 0
            print("WARNING: d2 resulta menor de 0")

        k = d/(d2-d)
        Qu = QL*(1+k)
        if verbose:
            print("k: ", k)
            print("Qu: ", Qu)

    BW=f0/QL;

    results_Kajfez = [f0, BW, QL, k, 0, Qu]

    return results, results_Kajfez, d, S11, S11off, d2
#####

#####
#Calculo de la Q_trans
def Qu_trans(f, S11, S21, S22, verbose):

    #verbose = 1 entonces se escriben los resultados intermedios de la funcion
    if verbose:
        print('TRANSMISSION MEASUREMENT')

    #pasa los parametros S a magnitud db y angulo
    f_MHz = f/1e6
    MS11 = np.absolute(S11);
    MS21 = np.absolute(S21);
    MS22 = np.absolute(S22);
    dBS11 = 20*np.log10(MS11);
    dBS21 = 20*np.log10(MS21);
    dBS22 = 20*np.log10(MS22);
    AS11 = np.angle(S11)*180/np.pi;
    AS21 = np.angle(S21)*180/np.pi;
    AS22 = np.angle(S22)*180/np.pi;

    if verbose:
        #realizamos metodo de los tres puntos
        print('Metodo de 3 puntos (-3dB)')

    #Se busca la frecuencia de resonancia
    dBS21d, indres = dBS21.max(0), dBS21.argmax(0)
    indd = indres
    f0 = f[indres]

    if verbose:
        print("dBS21: ", dBS21)
        print("dBS21d: ", dBS21d)

    #Busqueda de punto del ancho de banda 3 dB
    #primero se busca las frecuencias mas cercanas
    G3dB = dBS21d-3.01030
    dBS21d = np.array(dBS21[0:], dBS21[-1]);
    dBS21d = np.roll(dBS21, -1)
    LdBS21 = len(dBS21)
    LdBS21d = len(dBS21d)

    if verbose:

```

```

    print("LdBS21: ",LdBS21)
    print("LdBS21d: ",LdBS21d)
    print("\nG3dB: ",G3dB)

ind1 = np.where((dBS21 < G3dB) & (dBS21d > G3dB))
ind2 = np.where((dBS21 > G3dB) & (dBS21d < G3dB))

if verbose:
    print("\nind1: ",ind1)
    print("\nind2: ",ind2)

if len(ind1) > 1 or len(ind2) > 1:
    print('WARNING -> RIZADO ALTO: CONSIDERE AMPLIFICAR EL FILTRADO (Navg_pts)')

#se realiza la interpolacion
Af = f[1] - f[0]
Af = f[1] - f[0]
f1 = f[ind1[0]] + Af * (1 + ind1[-1]- ind1[0]) * (G3dB - dBS21[ind1[0]]) / (dBS21
[ind1[-1]+1] - dBS21[ind1[0]])
f2 = f[ind2[0]] + Af * (1 + ind2[-1]- ind2[0]) * (G3dB - dBS21[ind2[0]]) / (dBS21
[ind2[-1]+1] - dBS21[ind2[0]])

if verbose:
    print("f1: ",f1)
    print("f2: ",f2)
    print("\nAf: ",Af)
    print("f[ind1]: ",f[ind1])
    print("f[ind2]: ",f[ind2])

#Calculo del ancho de banda y de La Q
BW = abs(f2[0] - f1[0])
QL = f0 / BW
#Calculo de La Q aproximada
k = 0.5 * MS21[indres] / (1-MS21[indres])
Qu = QL / (1 - 10 ** (dBS21d / 20))

if verbose:
    print("BW: ",BW)
    print("QL: ",QL)
    print("k: ",k)
    print("Qu: ",Qu)

results1 = [f0, BW, QL, k, Qu]

if verbose:
    print("Resultados: ",results1)

#Calculo de La Qu considerando acoplos de entrada y de salida
VSWR1 = (1 + MS11[indres]) / (1 - MS11[indres])
VSWR2 = (1 + MS22[indres]) / (1 - MS22[indres])

if verbose:
    print("VSWR1: ",VSWR1)
    print("VSWR2: ",VSWR2)

#Se calcula Beta 1 y Beta 2 dependiendo de si esta infra o sobreacoplado
undercoupled1 = checkundercoupling(f, S11, verbose)

if verbose:
    print("undercoupled_trans1: ",undercoupled1)

if undercoupled1 == 1:
    beta1 = 1 / VSWR1
else:
    beta1 = VSWR1

```

```

undercoupled2 = checkundercoupling(f, S22, verbose)

if verbose:
    print("undercoupled_trans2: ",undercoupled2)

if undercoupled2 == 1:
    beta2 = 1 / VSWR2
else:
    beta2 = VSWR2

k1 = beta1
k2 = beta2
K1 = k1 * (1+k2) / (1 - k1 * k2)
K2 = k2 * (1+k1) / (1 - k1 * k2)
Qu = QL * (1 + K1 + K2)

if verbose:
    print("k1: ",k1)
    print("k2: ",k2)
    print("K1: ",K1)
    print("K2: ",K2)
    print("Qu: ",Qu)

results2= [f0, BW, QL, K1, K2, Qu]

if verbose:
    print("resultados finales: ",results2)

#Qu (trasnmission curve fitting (K. Leong)
#Comparing unloaded Q-factor of a high-Q dielectric resonator measured
#using the transmission mode and reflection mode methods involving
#S-parameter circle fitting", K. Leong, J. Mazierska, M.V. Jacob, D.
#Ledenyov, IEEE MTT-S International Microwave Symposium, 2002.

#min dBS21res, indres
#dBS21res, indres, S11res, f0, dBS21d, indd, S11d, fd = searchresonance(dBS21)
dBS11d,indres = dBS11.min(0),dBS11.argmin(0) #Poner en funcion de La de guti
f0 = f[indres]

if verbose:
    print("f0: ",f0)

# for con dos iteraciones para asegurar que los resultados son correctos
for i in [1,2]:

    f1 = f0
    #restrincion del Bw de evaluacion
    spank = 3*BW
    indk = np.where((f > (f1-spank/2)) & (f < (f1+spank/2)))
    fk = f[indk]
    S21k = S21[indk]
    t = 2 * (fk - f1)/f0

    if verbose:
        print("f1: ",f1)
        print("spank: ",spank)
        print("indk: ",indk)
        print("fk: ",fk)
        print("\nS21k: ",S21k)
        print("t: ",t)

    #Metodo TOMASO
    """
    A = [t, np.ones(len(t)), np.zeros(len(t)), np.zeros(len(t)), t * imag(S21k)]

```

```

print("Matriz A: ",A)

S21kreal = S21k.real
print("S21k real: ",S21kreal)

b = [S21kreal, imag(S21k)]
print("Matriz b: ",b)

b = b[:,]
print("Matriz b(:): ",b)

x = linalg.solve(A,b)
print("Resolucion sistema x = ",x)

a1 = x[1] + 1i * x[3]
a2 = x[2] + 1i * x[4]
a3 = 1i * x[5]
print("a1: ",a1)
print("a2: ",a2)
print("a3: ",a3)
"""

#Metodo PEDRO
A = [t, np.ones(len(fk)), -t*S21k]
At = np.transpose(A)

if verbose:
    print("Matriz A: ",A)
    print("Tamaño A: ",np.shape(A))
    print("Tamaño At: ",np.shape(At))

b = S21k
bt = S21k.transpose()

if verbose:
    print("Matriz b ",b)
    print("Matriz bt: ",bt)
    print("Tamaño B: ",np.shape(b))
    print("Tamaño Bt: ",np.shape(bt))

x = np.linalg.lstsq(At,b,rcond=None)

if verbose:
    print("Resolucion sistema x = ",x)

s = x[0]
a1 = s[0]
a2 = s[1]
a3 = s[2]

if verbose:
    print("a1: ",a1)
    print("a2: ",a2)
    print("a3: ",a3)

#Se realiza calculo de La Qu (QL de Leong)
Rhos = a1 / a3
QLL = a3.imag
d = abs(a2-a1/a3)

if verbose:
    print("Rhos: ",Rhos)
    print("QL: ",QL)
    print("QL: ",QL)
    print("d: ",d)

```

```

        print("\nt[0] = ",t[0])

#Traza del circulo con muchos puntos
t = np.linspace(t[0],t[-1],1601)
Rhoi=(a1*t+a2) / (a3*t+1)

if verbose:
    print("t: ",t)
    print("Rhoi: ",Rhoi)

#Se trabaja en dB en vez de con el Rhoi normal porque no se encuentra correctamente el max de Rhoi
MSRhoi = np.absolute(Rhoi)
dBRhoi = 20*np.log10(MSRhoi);

#Se vuelve a calcular la frecuencia de resonancia
S21resk,indresk = dBRhoi.max(0),dBRhoi.argmax(0)
tresk = t[indresk]
S21resk = Rhoi[indresk]
fres = tresk * f0/2 +f0
f0 = fres
BWL = f0 / QL

if verbose:
    print("\nS21resk: ",S21resk)
    print("indresk: ",indresk)
    print("tresk: ",tresk)
    print("fres: ",fres)
    print("BW: ",BWL)

# Si se cumplen los criterios que entre a calcular
# si no, poner valores a Los diametros para que calcule
# HacerLo por separado para cada uno de los puertos
if (max(dBS11) - min(dBS11)) > dBS11th:
    rqrefl11,rqrefl11K, dS11, S11, S11off, dl1 = Qu_refl(f, S11,"in", verbose)
else:
    dS11 = 0
    dl1 = 2
if (max(dBS22) - min(dBS22)) > dBS22th:
    rqrefl12,rqrefl12K, dS22, S22, S22off, dl2 = Qu_refl(f, S22,"in", verbose)
else:
    dS22 = 0
    dl2 = 2

if verbose:
    print("dS11: ",dS11)
    print("dS22: ",dS22)
    print("dl1: ",dl1)
    print("dl2: ",dl2)

beta2L = dS22 * dl1 / (dl2 * dl1 - dS22 * dl1 - dS11 * dl2)
beta1L= dS11 * dl2 / (dl2 * dl1 - dS22 * dl1 - dS11 * dl2)
K1L = beta1L;
K2L = beta2L;

if verbose:
    print("\nbeta2: ",beta2L)
    print("beta1: ",beta1L)
    print("K2: ",K2L)
    print("K1: ",K1L)
    print("S21resk = ", S21resk)

QuL = QLL * (1 + K1L + K2L)

```

```

if verbose:
    print("Qu: ",QuL)

results3 = [f0, BWL, QLL, K1L, K2L, QuL]

#Dibujamos La carta de smith
if Results:
    fig=plt.figure(9, figsize=(6,6))
    ax=plt.subplot(111)
    smithchart(S21,'b-', 'S21',Rhoi,'k-', 'Rhoi',ax)

    smithchart(rho1=S21resk,line_style1='c-o',label1='mo',ax=ax)

#devuelvo results
return results1, results2, results3
#####

#####

def checkundercoupling(f, S11, verbose):

    #pasa Los parametros S a magnitud db y angulo
    f_MHz = f/1e6
    MS11 = np.absolute(S11)
    dBS11 = 20*np.log10(MS11)
    AS11 = np.angle(S11)*180/np.pi

    # Busqueda de resonancia de S11 y valor desintonizado
    dBS11res, indres = dBS11.min(), dBS11.argmin() # Valor y posicion
    dBS11d, indd = dBS11.max(), dBS11.argmax() #Valor y posicion

    # Phase offset del círculo Q a la recta real
    S11off = phaseoffset(f, S11, verbose)

    S11offreal = S11off.real

    # Determinamos si es under/overcoupled
    if S11offreal[indres] <= 0:
        undercoupled = 1;
    else:
        undercoupled = 0;

    return undercoupled
#####

#####

def phaseoffset(f, S11,verbose):

    #pasa Los parametros S a magnitud db y angulo
    f_MHz = f/1e6
    MS11 = np.absolute(S11)
    dBS11 = 20*np.log10(MS11)
    AS11 = np.angle(S11)*180/np.pi

    # Busqueda de resonancia de S11
    dBS11res, indres = dBS11.min(), dBS11.argmin() # Valor y posicion
    S11res = S11[indres]
    f0 = f[indres]

    # Busqueda de S11 desintonizado
    dBS11d, indd = dBS11.max(), dBS11.argmax() #Valor y posicion
    S11d = S11[indd]

```

```

    if verbose:
        print('S11d '+str(S11d))

    MS11d = max(np.abs(S11[0]),np.abs(S11[-1]))
    AS11d = np.angle(S11[0] + S11[-1])
    S11d = MS11d * np.exp(1j*AS11d)

    if verbose:
        print('S11d '+str(S11d))

    phaseS11d = np.angle(S11d)*180/np.pi

    if verbose:
        print(phaseS11d)

    phaseS11res = np.angle(S11res)*180/np.pi

    VSWRres = (1 + abs(S11res))/(1 - abs(S11res))
    if VSWRres > 1.15 : # /Rho/>0.07 (Usamos S11res)
        if abs(np.angle(S11d/S11res)*180/np.pi) < 40 :
            phaseobj = 180
        else:
            phaseobj = 360
        S11off = S11 * np.exp(1j*np.pi/180*(phaseobj-phaseS11res))
    if VSWRres <= 1.15 : # /Rho/<0.07 (Usamos S11d)
        phaseobj = 180
        S11off = S11 * np.exp(1j*np.pi/180*(phaseobj-phaseS11d))

    return S11off
#####

#####

def Qu_r_t_correction(f01, QL1, k1, f02, QL2, k2, d1, d11, d2, d12, verbose):

    K1 = k1 * (1 + k2)/(1 - k1 * k2);
    K2 = k2 * (1 + k1)/(1 - k1 * k2);
    Qu1 = QL1 * (1 + K1 + K2);
    Qu2 = QL2 * (1 + K1 + K2);
    BW1 = f01/QL1;
    BW2 = f02/QL2;

    if verbose:
        print("\nLa Qu correction\n")
        print("k1: ", k1)
        print("k2: ", k2)
        print("K1: ",K1)
        print("K2: ",K2)
        print("Qu1: ",Qu1)
        print("Qu2: ",Qu2)
        print("BW1: ",BW1)
        print("BW2: ",BW2)

    results = [f01, f02, BW1, BW2, QL1, QL2, K1, K2, Qu1, Qu2];

    return results
#####

#####

def searchresonance(f, S11, dBS11, verbose):

    #####AÑADIR para trans o para reflex#####

```

```

# Función que busca la resonancia de S11 y el valor desintonizado

#####Cuidado al calcular el max y min de S11, no parece que sea el máximo del va
lor absoluto de S11, como Matlab
#####Pasado a dB sí que va bien

dBS11res, indres = dBS11.min(), dBS11.argmin() # Valor y posicion
S11res = S11[indres]
f0 = f[indres]
if verbose:
    print()
    print('dBS11 at resonance: '+str(dBS11res))
    print('S11 at resonance: '+str(S11res))
    print('index at resonance: '+str(indres))
    print('f0: '+str(f0))

# Búsqueda de S11 desintonizado
dBS11d, indd = dBS11.max(), dBS11.argmax() #Valor y posicion
S11d = S11[indd]
fd = f[indd];

if verbose:
    print()
    print('dBS11 detuned: '+str(dBS11d))
    print('S11 detuned: '+str(S11d))
    print('index of detuned S11: '+str(indd))
    print('f detuned: '+str(fd))
    print()

# Búsqueda de S11 desintonizado (alternativa)
dBS11d, indd = dBS11.max(), dBS11.argmax() #Valor y posicion
MS11d = max(np.abs(S11[0]), np.abs(S11[-1]))
AS11d = np.angle(S11[0] + S11[-1])
S11d = MS11d * np.exp(1j*AS11d)
fd = f[indd];

if verbose:
    print()
    print('dBS11 detuned: '+str(dBS11d))
    print('S11 detuned: '+str(S11d))
    print('index of detuned S11: '+str(indd))
    print('f detuned: '+str(fd))
    print()

#The default in numpy (arr.max() or np.max(arr) when arr is complex) is the compl
ex value with the maximum real part.

return dBS11res, indres, S11res, f0, dBS11d, indd, S11d, fd
#####

#####
def Calculate(f, S11, S21, S12, S22, verbose):

    global fig, dBS11th, dBS22th

    #tenemos los parametros S en arrays, se comienza con los calculos
    res_txt = []
    results = []
    k_r1 = 0
    QL_r1 = 0
    resultados=datetime.datetime.now().strftime("%Y_%m_%d__%H_%M_%S_Resultados.txt" )

    #archivo de escritura de resultados

```



```

fichero_resultados=open(resultados,"w")

measRef1P1 = 0
measRef1P2 = 0
measTx = 0

#Genero los umbrales para realizar las comparaciones
dBS11th = 0.2
dBS22th = 0.2
dBS21th = -70

#pasa los parametros S a magnitud db y angulo --> Hacer función para pasar datos
f_MHz = f/1e6
MS11 = np.absolute(S11);      MS21 = np.absolute(S21)
MS12 = np.absolute(S12);      MS22 = np.absolute(S22)
dBS11 = 20*np.log10(MS11);     dBS21 = 20*np.log10(MS21)
dBS12 = 20*np.log10(MS12);     dBS22 = 20*np.log10(MS22)
AS11 = np.angle(S11)*180/np.pi; AS21 = np.angle(S21)*180/np.pi
AS12 = np.angle(S12)*180/np.pi; AS22 = np.angle(S22)*180/np.pi

#####
# medida de reflexion en el puerto 1
# Se comprueba si al menos hay un pico de resonancia el S11
if (max(dBS11) - min(dBS11)) > dBS11th:
    measRef1P1 = 1

    results11, resultsk1, d1, S11, S11off, d11 = Qu_refl(f, S11, "in", verbose)

    print("\nLos resultados en reflexion, para el puerto 1, son los siguientes:")
    print("\nMétodo de los 3 puntos:\nf0 = ", round(results11[0]/1e6, 6),"MHz", "
\nBW = ",round(results11[1]/1e3,3),"KHz", "\nQL = ", round(results11[2],0), "\nbeta1
= ", round(results11[3],4), "\nbeta2 = 0 \nQu = ",round(results11[5],0))
    print("\nMétodo de kajfez:\nf0 = ", round(resultsk1[0]/1e6,6),"MHz", "\nBW =
",round(resultsk1[1]/1e3,3),"KHz", "\nQL = ", round(resultsk1[2],0), "\nbeta1 = ", ro
und(resultsk1[3],4), "\nbeta2 = 0 \nQu = ",round(resultsk1[5],0))

    results_opt = resultsk1; # Quedarse con la beta más grande en vez de hacerlo
así

    if Results:
        fig=plt.figure(7, figsize=(6,6))
        ax=plt.subplot(111)
        plt.title('S11 y S11 con offset')
        smithchart(S11,'b-', 'S11',S11off,'r-', 'S11off',ax)

        texto1 = ["f0 = %s" % results11[0],", BW = %s" % results11[1],", QL = %s" % r
esults11[2],", beta1 = %s"% results11[3],", beta2 = 0",", Qu = %s"% results11[5]]
        texto2 = ["f0 = %s" % resultsk1[0],", BW = %s" % resultsk1[1],", QL = %s" % r
esultsk1[2],", beta1 = %s"% resultsk1[3],", beta2 = 0",", Qu = %s"% resultsk1[5]]

        fichero_resultados.write("\nResultados en reflexion en el puerto 1: ")

        fichero_resultados.write("\nMétodo de los 3 puntos: ")
        for i in [0, 1, 2, 3, 4, 5] :
            fichero_resultados.write(texto1[i])

        fichero_resultados.write("\nMétodo de Kajfez: ")
        for i in [0, 1, 2, 3, 4, 5] :
            fichero_resultados.write(texto2[i])
    else:
        print("\nNo se pueden dar valores para reflexión desde el puerto 1\n")
#####

```

```
#####
#medida de reflexion en el puerto 2
# Para que entre que la diferencia entre el máximo y el mínimo sea mayor de 0.3dB
if (max(dBS22) - min(dBS22)) > dBS22th:
    measRef1P2 = 1

    results12, resultsk2, d2, S22, S22off, dl2 = Qu_refl(f, S22, "out", verbose)

    print("\nLos resultados en reflexion, para el puerto 2, son los siguientes:")
    print("\nMétodo de los 3 puntos:\nf0 = ", round(results12[0]/1e6,6),"MHz", "\nBW = ", round(results12[1]/1e3,3),"kHz", "\nQL = ", round(results12[2],0), "\nbeta1 = 0 \nbeta2 = ", round(results12[3],4), "\nQu = ", round(results12[5],0))
    print("\nMétodo de kajfez:\nf0 = ", round(resultsk2[0]/1e6,6),"MHz", "\nBW = ", round(resultsk2[1]/1e3,3),"kHz", "\nQL = ", round(resultsk2[2],0), "\nbeta1 = 0 \nbeta2 = ", round(resultsk2[3],4), "\nQu = ", round(resultsk2[5],0))

    #results_opt = resultsk2;

    if Results:
        fig=plt.figure(8, figsize=(6,6))
        ax=plt.subplot(111)
        plt.title('S22 y S22 con offset')
        smithchart(S22, 'b-', 'S22', S22off, 'r-', 'S22off', ax)

        #Escribir en un fichero los resultados
        texto1 = ["f0 = %s" % results12[0],",", "BW = %s" % results12[1],",", "QL = %s" % results12[2],",", "beta1 = 0",",", "beta2 = %s" % results12[3],",", "Qu = %s" % results12[5]]
        texto2 = ["f0 = %s" % resultsk2[0],",", "BW = %s" % resultsk2[1],",", "QL = %s" % resultsk2[2],",", "beta1 = 0",",", "beta2 = %s" % resultsk2[3],",", "Qu = %s" % resultsk2[5]]

        fichero_resultados.write("\n\nResultados en reflexion en el puerto 2: ")

        fichero_resultados.write("\nMétodo de los 3 puntos: ")
        for i in [0, 1, 2, 3, 4, 5] :
            fichero_resultados.write(texto1[i])

        fichero_resultados.write("\nMétodo de Kajfez: ")
        for i in [0, 1, 2, 3, 4, 5] :
            fichero_resultados.write(texto2[i])
    else:
        print("\nNo se pueden dar valores para reflexión desde el puerto 2\n")
#####

##### REFLECTION MEASUREMENT (TRANSMISION RESONATOR) #####

if (max(dBS11) - min(dBS11)) > dBS11th and (max(dBS22) - min(dBS22)) > dBS22th and max(dBS21) > dBS21th:

    f01 = resultsk1[0]; QL1 = resultsk1[2]; k1 = resultsk1[3];
    f02 = resultsk2[0]; QL2 = resultsk2[2]; k2 = resultsk2[3];

    #comprobar dL2 y dL1 no estan usandose
    results1 = Qu_r_t_correction(f01,QL1,k1,f02,QL2,k2,d1,dl1, d2,dl2,verbose);

    print("\nResultados en reflexion con dos puertos: ")
    print("\nResultados reflexion 1, dos puertos:\nf0= ",round(results1[0]/1e6,6),"MHz", "\nBW= ",round(results1[2]/1e3,3),"kHz", "\nQL= ",round(results1[4],0), "\nbeta1 = ",round(results1[6],4), "\nbeta2= ",round(results1[7],4), "\nQu= ",round(results1[8],0))
    print("\nResultados reflexion 2, dos puertos:\nf0= ",round(results1[1]/1e6,6),"MHz", "\nBW= ",round(results1[3]/1e3,3),"kHz", "\nQL= ",round(results1[5],0), "\nbeta1 = ",round(results1[6],4), "\nbeta2= ",round(results1[7],4), "\nQu= ",round(results1[9],0))

```

```

        texto1 = ["f0 = %s" % results1[0],",", BW = %s" % results1[2],",", QL = %s" % results1[4],",", beta1 = %s" % results1[6],",", beta2 = %s" % results1[7],",", Qu = %s" % results1[8]]
        texto2 = ["f0 = %s" % results1[1],",", BW = %s" % results1[3],",", QL = %s" % results1[5],",", beta1 = %s" % results1[6],",", beta2 = %s" % results1[7],",", Qu = %s" % results1[9]]

        fichero_resultados.write("\n\nResultados en reflexion con dos puertos: ")

        fichero_resultados.write("\nResultados reflexion 1, dos puertos: ")
        for i in [0, 1, 2, 3, 4, 5] :
            fichero_resultados.write(texto1[i])

        fichero_resultados.write("\nResultados reflexion 2, dos puertos: ")
        for i in [0, 1, 2, 3, 4, 5] :
            fichero_resultados.write(texto2[i])
    else:
        print("\nNo se puede realizar la corrección para los valores de los dos puertos\n")

#####

#Medida de transmision
if max(dBS21) > dBS21th: #al menos un pico de resonancia
    measTx = 1
    results2, results3, results4 = Qu_trans(f, S11, S21, S22, verbose)

    print("\nLos resultados en transmision son los siguientes: ")
    print("\nMétodos de los 3 puntos:\nf0= ",round(results3[0]/1e6,6),"MHz","\nBW= ",round(results3[1]/1e3,3),"KHz","\nQL= ",round(results3[2],0),"\nbeta1= ",round(results3[3],4),"\nbeta2= ",round(results3[4],4),"\nQu= ",round(results3[5],0))
    print("\nMétodos de K. Leong:\nf0= ",round(results4[0]/1e6,6),"MHz","\nBW= ",round(results4[1]/1e3,3),"KHz","\nQL= ",round(results4[2],0),"\nbeta1= ",round(results4[3],4),"\nbeta2= ",round(results4[4],4),"\nQu= ",round(results4[5],0))

    results_opt = results4;

    #Escribir en un fichero los resultados
    texto1 = ["f0 = %s" % results3[0],",", BW = %s" % results3[1],",", QL = %s" % results3[2],",", beta1 = %s" % results3[3],",", beta2 = %s" % results3[4],",", Qu = %s" % results3[5]]
    texto2 = ["f0 = %s" % results4[0],",", BW = %s" % results4[1],",", QL = %s" % results4[2],",", beta1 = %s" % results4[3],",", beta2 = %s" % results4[4],",", Qu = %s" % results4[5]]

    fichero_resultados.write("\n\nResultados en transmision: ")

    fichero_resultados.write("\nMétodo de los 3 puntos: ")
    for i in [0, 1, 2, 3, 4, 5] :
        fichero_resultados.write(texto1[i])

    fichero_resultados.write("\nMétodo de K. Leong: ")
    for i in [0, 1, 2, 3, 4, 5] :
        fichero_resultados.write(texto2[i])

    #Se cierra fichero
    fichero_resultados.close()

    return results_opt
#####

```

```
#####
# DEFINING A GUI TO VIEW RESULTS
# Generate GUI

import ipywidgets as widgets
from ipywidgets import Layout
from IPython.display import display
from tkinter import Tk, filedialog
from IPython.display import clear_output, display
import datetime
import time
from ipywidgets import Image

Verbose = False
Results = False

class Qo_Factor_VNA_GUI():

    def __init__(self):

        # TITLE
        titulo = widgets.Label(value = 'ESS BILBAO: QUALITY FACTOR MEASUREMENT')
        box_tit = widgets.HBox(children=[titulo], layout = Layout(width = '100%'))

        #Logos
        with open('ess.jpeg','rb') as f:
            logo = f.read()
            logo1 = Image(value=logo, layout = Layout(width = '100px'))

        with open('uc.jpeg','rb') as f:
            logo = f.read()
            logo2 = Image(value=logo, layout = Layout(width = '50px'))

        box_imagen1 = widgets.HBox(children=[logo1], layout = Layout(width = '60%'))
        box_imagen2 = widgets.HBox(children=[logo2], layout = Layout(width = '100%'))

        box_titulo = widgets.HBox(children = [box_imagen1, box_tit, box_imagen2], layout = Layout(width = '100%'))

        # Elección para leer parametros de entrada (VNA o fichero s2p)
        rdbtn_VNA = widgets.RadioButtons(options = ['VNA', 'Input File'],
                                         value = 'Input File', disabled = False)

        txt_File = widgets.Text(value = '', description = 'File: ', disabled = False)
        btn_browse = widgets.Button(description = 'Browse', disabled = False,
                                     button_style = 'success', tooltip = 'Clic to select the file', icon = '')

        rdbtn_parameters = widgets.RadioButtons(options =
            ['Manualmente', 'Ordenador'], value = 'Manualmente', disabled = False)
        rdbtn_parameters.layout.visibility = 'hidden'

        titulo1 = widgets.Label(value = 'Input Data')
        titulo2 = widgets.Label(value = 'Introducir parámetros')
        titulo2.layout.visibility = 'hidden'

        box_input_data = widgets.HBox(children = [titulo1, rdbtn_VNA, txt_File, btn_browse], layout = Layout(width = '100%', border = 'solid 2px'))

        titulo3 = widgets.Label(value = 'Output Data')
        txt_f0 = widgets.FloatText(description = 'f0 (MHz): ', disabled = False)
        txt_Qo = widgets.FloatText(description = 'Qo: ', disabled = False)
        txt_Ql = widgets.FloatText(description = 'Ql: ', disabled = False)
        txt_b1 = widgets.FloatText(description = 'beta1: ', disabled = False)
```

```

txt_b2 = widgets.FloatText(description = 'beta2: ', disabled = False)

ch_Show = widgets.Checkbox(value=Results, description = 'Show Results',
                             disabled=False)
ch_History = widgets.Checkbox(value=False, description = 'History')
ch_Verbose = widgets.Checkbox(value=Verbose, description = 'Verbose',
                               disabled=False)

# Boton para guardar resultados con nombre a elegir y fecha

btn_Calculate = widgets.Button(description = 'Calculate', disabled = False,
                                button_style = 'success', tooltip = 'Clic to calculate the results',
                                icon = '')

box_calculate = widgets.VBox(children = [ch_Show, ch_History, ch_Verbose])

box_results = widgets.HBox(children = [box_calculate, btn_Calculate],
                             layout = Layout(border = 'solid 2px', width = '50%'))

box_output_data = widgets.VBox(children = [titulo3, txt_f0, txt_Q0, txt_Q1, t
xt_b1, txt_b2], layout = Layout(border = 'solid 2px', width = '50%'))

box_final = widgets.HBox(children = [box_results, box_output_data])

#box_titulo = widgets.HBox(children = [titulo],
#layout = Layout(border = 'solid 2px'))

box_complete = widgets.VBox(children=[box_titulo, box_input_data, box_final],
                              layout = Layout(border = 'solid 2px'))

display(box_complete)
out = widgets.Output()

def Radio_Button(change):

    if rdbtn_VNA.value == 'VNA':
        btn_browse.button_style = 'danger'
        btn_browse.disabled = True
        txt_File.disabled = True
        rdbtn_parameters.layout.visibility = 'visible'
        titulo2.layout.visibility = 'visible'

    if rdbtn_VNA.value == 'Input File':
        btn_browse.button_style = 'success'
        btn_browse.disabled = False
        txt_File.disabled = False
        rdbtn_parameters.layout.visibility = 'hidden'
        titulo2.layout.visibility = 'hidden'

rdbtn_VNA.observe(Radio_Button)

def Browse_Button(b):
    global fichero
    #clear_output()
    root = Tk()
    root.withdraw() # Hide the main window.
    root.call('wm', 'attributes', '.', '-topmost', True) # Raise the root to
the top of all windows.
    b.files = filedialog.askopenfilename(title = 'Select a File') # List of s
elected files will be set button's file attribute.
    texto = b.files.split('/')
    fichero = b
    file = texto[-1]
    txt_File.value = file

```

```

btn_browse.on_click(Browse_Button)

def Check_Verbose(change):
    global Verbose
    Verbose = ch_Verbose.value
    #print(Verbose)
    #print(change)

ch_Verbose.observe(Check_Verbose)

def Show_Results(change):
    global Results
    Results = ch_Show.value

ch_Show.observe(Show_Results)

def Calculate_Button(change):
    if rdbtn_VNA.value == 'VNA':
        f, S11, S21, S12, S22 = PNA()

    if rdbtn_VNA.value == 'Input File':
        f, S11, S21, S12, S22 = Call_File(fichero)

    results_opt = Calculate(f, S11, S21, S12, S22, Verbose)
    txt_f0.value = round(results_opt[0]/1e6, 6)
    txt_Qo.value = round(results_opt[5], 0) # 0 decimales para las Q's
    txt_Ql.value = round(results_opt[2], 0)
    txt_b1.value = round(results_opt[3], 4) # 4 decimales para b1 y b2
    txt_b2.value = round(results_opt[4], 4)

    btn_Calculate.on_click(Calculate_Button)
#####

#####
# Run GUI
QoF = Qo_Factor_VNA_GUI()
#####

```